BASES DE DATOS

PRÁCTICA 3

ÁLVARO FRAIDIAS MONTEAGUDO

2020

Índice

[CONTEXTO DE LA PRÁCTICA 2](#_Toc38296488)

[DISEÑO CONCEPTUAL 2](#_Toc38296489)

[DIFERENCIAS RESPECTO A LA BD PROPORCIONADA 2](#_Toc38296490)

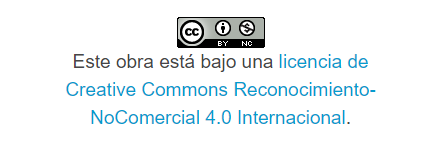
[DISEÑO LÓGICO 3](#_Toc38296491)

[CUESTIONES PREVIAS 4](#_Toc38296492)

[LISTA DE CONSULTAS 5](#_Toc38296493)

[TAREAS DE ADMINISTRACIÓN 17](#_Toc38296494)

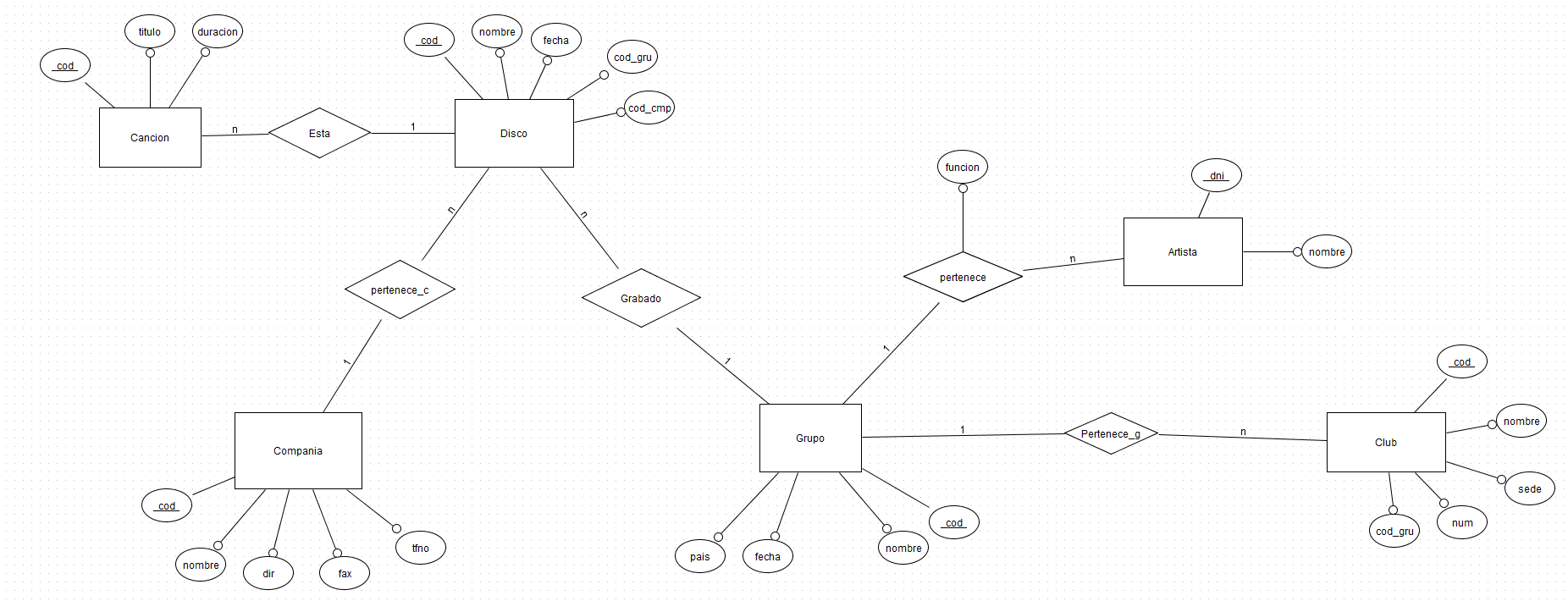
[REFERENCIAS BIBLIOGRÁFICAS 17](#_Toc38296495)



# CONTEXTO DE LA PRÁCTICA

En esta práctica empezamos a crear consultas SQL además de seguir practicando el diseño conceptual y lógico de una base de datos.

# DISEÑO CONCEPTUAL



# DIFERENCIAS RESPECTO A LA BD PROPORCIONADA

He tenido que crear varias relaciones para poder hacer el diseño conceptual ya que en la base de datos de Access había relaciones mediante claves foráneas. Una clave foránea son referencias a registros de otra tabla.

Las relaciones que he tenido que añadir son las siguientes:

**\*Pertenece\_c:** Se encarga de relacionar la entidad Disco con la entidad Companyia. Es una relación 1:N ya que puede haber muchos discos en una companyia. Los atributos que identifican a cada entidad en la relación son: Cod\_comp hace referencia a Disco y cod hace referencia a Companyia.

**\*Grabado:** Esta relación une la entidad Disco con su grupo. Es una relación 1:N ya que puede haber muchos discos grabados por un grupo. Los atributos que identifican a cada entidad en la relación son: Cod\_gru hace referencia a Disco y cod hace referencia a Grupo.

**\*Pertenece\_g:** Se encarga de relacionar la entidad Grupo con la entidad Club. Es una relación 1:N ya que puede haber muchos clubes de fans para un mismo grupo. Los atributos que identifican a cada entidad en la relación son: Cod hace referencia a Grupo y cod\_gru hace referencia a Club.

# DISEÑO LÓGICO

/\* Entidad Cancion: \*/

Cancion (cod.cancion: dom\_cod.cancion, titulo:dom\_titulo, duracion:dom\_duracion )

CP { cod.cancion }

VNN { titulo }

VNN{ duracion }

/\* Entidad Compania: \*/

Compania (cod.comp: dom\_cod.comp, nombre: dom\_nombre, dir: dom\_dir, fax: dom\_fax, tfno: dom\_tfno, cod.disco: dom\_cod.disco)

CP { cod.comp }

VNN { nombre }

VNN { dir }

VNN { fax }

VNN { tfno }

VNN { cod.disco }

CAj:{cod.disco} hace referencia a Disco

/\* Entidad Club: \*/

Club (cod.club: dom\_cod.club, nombre: dom\_nombre, sede: dom\_sede, num: dom\_num, cod\_gru: dom\_cod\_gru)

CP { cod.club }

VNN { nombre }

VNN{ sede }

VNN { num }

VNN { cod\_gru }

/\* Entidad Grupo: \*/

Grupo (cod.grupo: dom\_cod.grupo, nombre: dom\_nombre, fecha: dom\_fecha, pais: dom\_pais, cod.disco: dom\_cod.disco, cod.club: dom\_cod.club )

CP{ cod.grupo }

VNN { nombre }

VNN{ fecha }

VNN { pais }

VNN { cod.disco }

CAj:{cod.disco} hace referencia a Disco

VNN { cod.club }

CAj:{cod.club} hace referencia a club

/\* Entidad Artista: \*/

Artista (dni: dom\_dni, nombre: dom\_nombre, cod.grupo: dom\_cod.grupo)

CP { dni }

VNN { nombre }

VNN{ funcion }

VNN { cod.grupo }

CAj:{cod.grupo} hace referencia a Grupo

/\* Entidad Disco: \*/

Disco (cod.disco: dom\_cod.disco, nombre: dom\_nombre, fecha: dom\_fecha, cod.cancion: dom\_cod.cancion )

CP { cod.disco }

VNN{ nombre }

VNN { fecha }

VNN { cod.cancion }

CAj: {cod.cancion} hace referencia a Canción

# CUESTIONES PREVIAS

**1. ¿Qué tipo de operación: a) producto cartesiano o b) concatenación, se está realizando en la cláusula WHERE tanto a través de la igualdad (d.cod\_gru=g.cod\_grupo) como del predicado IN?**

Es un producto cartesiano ya que compara todas las columnas de disco con todas las columnas de grupo que sus códigos sean iguales.

**2. Proponer otra solución haciendo uso del predicado EXISTS**

SELECT d.nombre

FROM Disco d

WHERE EXIST (SELECT g.cod\_grupo

FROM Grupo g

WHERE d.cod\_gru = g.cod\_grupo AND g.nombre = 'U2')

**3. ¿Por qué predicado, de los estudiados en la parte II del SQL, podrías sustituir el igual en esta parte de la sentencia: g.nombre = ‘U2’?**

Por el predicado LIKE

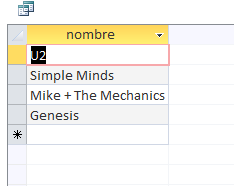
# LISTA DE CONSULTAS

1 Obtener el nombre de los grupos de música que no sean de España.

SELECT grupo.nombre

FROM grupo

WHERE grupo.pais NOT LIKE 'España';

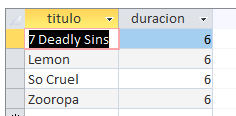


2 Seleccionar sólo el título de todas las canciones que duren más de 5 minutos.

SELECT cancion.titulo, cancion.duracion

FROM cancion

WHERE cancion.duracion > 5;



3 Obtener el número total de discos de la base de datos

SELECT count(disco.nombre)

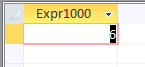
FROM disco;



4 Obtener cuántos minutos dura la canción más larga.

SELECT max(canción.duracion)

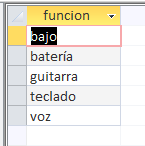
FROM canción;



5 Obtener las distintas funciones que un artista puede desempeñar dentro de un grupo.

SELECT DISTINCT pertenece.funcion

FROM pertenece;

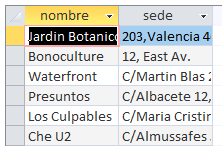


6 Selecciona tanto el nombre como la sede de los clubes de fans con un número de socios entre 100 y 500.

SELECT club.nombre, club.sede

FROM club

WHERE num between 100 and 500;

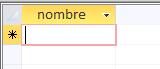


7 Obtener los nombres de los clubes de fans que no tienen definida la dirección de su sede o el número de admiradores

SELECT club.nombre

FROM club

WHERE club.sede IS NULL OR club.num IS NULL;

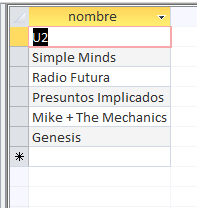


8 Obtener los nombres de los grupos por orden alfabético inverso de nombre (de la Z a la A).

SELECT grupo.nombre

FROM grupo

ORDER BY grupo.nombre DESC;

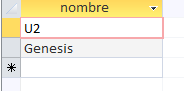


9. Obtener el nombre de los grupos que sólo constan de una palabra

SELECT nombre

FROM grupo

WHERE nombre NOT LIKE "[a-z]\* \*";

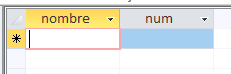


10 Obtener el nombre y número de integrantes de los clubes de fans cuyo nombre contiene la cadena “100%”

SELECT club.nombre, club.num

FROM club

WHERE club.nombre IN ('100%');

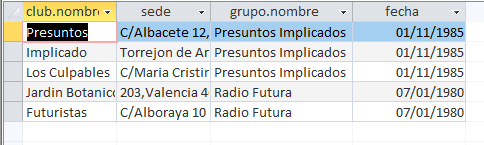


11 Seleccionar nombre y sede de los clubes de fans de grupos de España, así como el nombre y fecha de formación del grupo al que admiran.

SELECT club.nombre, club.sede, grupo.nombre, grupo.fecha

FROM club, grupo

WHERE club.cod\_gru = grupo.cod AND grupo.pais = 'España';

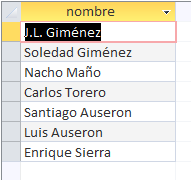


12 Seleccionar el nombre de todos los artistas que pertenezcan a grupos españoles.

SELECT artista.nombre

FROM artista, pertenece, grupo

WHERE artista.dni = pertenece.dni AND grupo.cod = pertenece.cod AND grupo.pais = 'España';

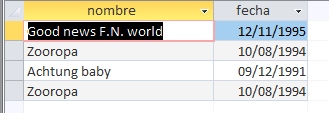


13 Seleccionar los discos (nombre y fecha de publicación) que contengan alguna canción con una duración que sobrepase los 5 minutos.

SELECT disco.nombre, disco.fecha

FROM disco, cancion, esta

WHERE disco.cod = esta.cod AND cancion.cod = esta.can AND cancion.duracion > 5;



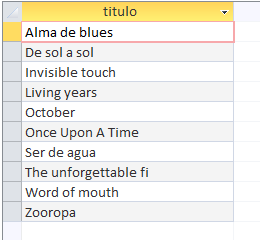
14 Obtener los nombres de las canciones que dan nombre al disco en el que aparecen, ordenadas alfabéticamente.

SELECT cancion.titulo

FROM disco, cancion, esta

WHERE disco.cod = esta.cod AND cancion.cod = esta.can AND disco.nombre = cancion.titulo;

ORDER BY canción.titulo



15 Obtener los nombres y direcciones de las compañías que han grabado algún disco cuyo nombre empiece por ‘A’ publicado durante la década de los 80 (1980-1989).

SELECT c.nombre, c.dir

FROM companyia c, disco d

WHERE c.cod IN (d.cod\_comp) AND

d.nombre LIKE 'A%' AND

d.fecha BETWEEN 1980 AND 1989;

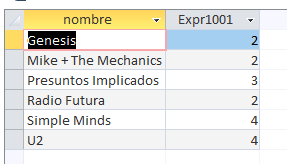
16 Obtener cuántos clubs de fans tiene cada grupo, junto con el nombre del grupo

SELECT grupo.nombre, count(club.nombre)

FROM grupo, club

WHERE club.cod\_gru = grupo.cod

GROUP BY grupo.nombre;



17 Seleccionar el nombre de los discos que han sido grabados después de 1986 por grupos que tengan algún club de fans con más de 5000 socios.

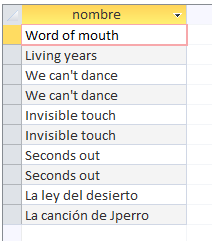
SELECT disco.nombre

FROM disco, grupo, club

WHERE disco.cod\_gru = grupo.cod

AND grupo.cod = club.cod\_gru

AND club.num > 5000 AND disco.fecha > 1986;

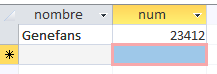


18. Seleccionar el nombre del club de fans que tiene más integrantes, indicando además cuántos fans lo componen.

SELECT club.nombre,club.num

FROM club

WHERE club.num = (SELECT MAX(num) FROM club);

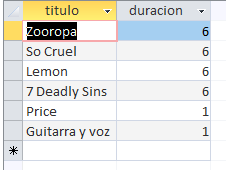


19. Seleccionar los títulos de las canciones tanto de mayor duración como de menor duración, indicando cuánto dura cada una.

SELECT titulo,duracion

FROM cancion

WHERE cancion.duracion = (SELECT max(duracion) FROM cancion) OR (SELECT min(duracion) FROM cancion)



20. Seleccionar el número de DNI y el nombre de los artistas que han pertenecido a más de un grupo de los registrados en la base de datos.

SELECT DISTINCT a.dni, a.nombre

FROM artista a, pertenece p

WHERE a.dni = p.dni

AND EXISTS (SELECT \*

FROM pertenece

WHERE a.dni=dni and cod<>p.cod)

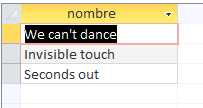


21. Seleccionar el nombre de todos los discos del grupo que fue fundado hace más tiempo.

SELECT disco.nombre

FROM disco,grupo

WHERE disco.cod\_gru IN (grupo.cod) AND grupo.fecha = (SELECT fffffffffffffffffffffffffffffffffffffffffffffffffffffffmin(grupo.fecha) fffffffffffffffffffffffffffffffffffffffffffffffffffffffFROM grupo);



22. Seleccionar el código, nombre y dirección de todas las compañías discográficas que no han grabado ningún disco de grupos ingleses.

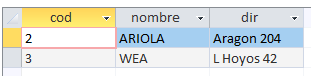
SELECT DISTINCT companyia.cod, companyia.nombre, companyia.dir

FROM companyia, disco, grupo

WHERE companyia.cod = disco.cod\_comp

AND disco.cod\_gru = grupo.cod

AND grupo.pais NOT LIKE 'Inglaterra';



23. Obtener el nombre de las compañías discográficas que sólo han publicado trabajos de grupos ingleses.

SELECT DISTINCT companyia.nombre

FROM companyia, disco, grupo

WHERE companyia.cod = disco.cod\_comp

AND disco.cod\_gru = grupo.cod

AND grupo.pais LIKE 'Inglaterra';



24. Seleccionar nombre y número de teléfono de aquellas compañías discográficas que hayan grabado todos los discos de alguno de los grupos.

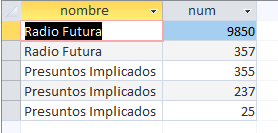
25. Obtener los nombres de los grupos en los que todos los integrantes tienen diferentes funciones.

26. Seleccionar el número total de fans de cada grupo español, además de su nombre, ordenados de mayor a menor número de fans. a. OPCIONAL: modificar la consulta para que SELECT grupo.nombre, club.num

FROM grupo, club

WHERE grupo.cod = club.cod\_gru AND grupo.pais LIKE 'España' OR club.num is null

ORDER BY club.num DESC;



27. Seleccionar los grupos (nombre y número total de componentes) formados por más de tres artistas.

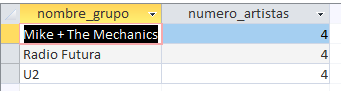
SELECT grupo.nombre AS nombre\_grupo, count(artista.dni) AS numero\_artistas

FROM grupo, pertenece, artista

WHERE grupo.cod = pertenece.cod AND pertenece.dni = artista.dni

GROUP BY (grupo.nombre)

HAVING count(artista.dni) > 3



28. Obtener cuántas canciones han sido grabadas por cada compañía discográfica, junto con el nombre y la dirección de la compañía.

SELECT companyia.nombre, companyia.dir, COUNT(esta.can) AS numero\_canciones

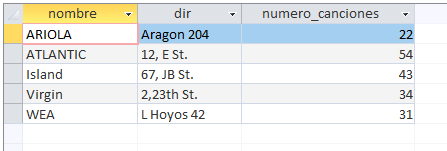
FROM esta, companyia, cancion, disco

WHERE companyia.cod = disco.cod\_comp AND

disco.cod = esta.cod AND

esta.can = cancion.cod

GROUP BY companyia.nombre, companyia.dir;



29. Obtener cuántos discos han sido grabados por los grupos que tienen más de 1000 fans (sumando los de todos sus clubes de admiradores), junto con el nombre del grupo.

SELECT g.nombre, COUNT (d.nombre) AS numero\_discos

FROM grupo g, disco d

WHERE g.cod IN (d.cod\_gru) AND g.cod IN (

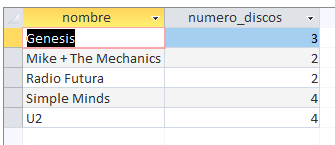
SELECT cod\_gru

FROM club

GROUP BY cod\_gru,num

HAVING SUM (num) > 1000)

GROUP BY g.nombre;



30. Seleccionar los títulos y duración de todas las canciones grabadas por el grupo al que admira el club de fans con menor número de integrantes.

SELECT DISTINCT cancion.titulo, cancion.duracion

FROM cancion

WHERE cancion.cod IN (SELECT can

FROM esta

WHERE cod IN (SELECT cod

FROM disco

WHERE cod\_gru IN (SELECT grupo.cod

FROM grupo,club

WHERE grupo.cod IN (club.cod\_gru)

AND num IN (SELECT MIN(num) FROM eee club))));



31. Seleccionar el título y duración de la canción más larga, pero sólo si es la única con esa duración.

SELECT cancion.titulo, cancion.duracion

FROM cancion

WHERE cancion.duracion = (SELECT DISTINCT MAX(duracion)

FROM cancion

GROUP BY duracion

HAVING COUNT (duracion) = 1)



32. Seleccionar el sexto club con mayor número de fans (es decir, que haya cinco clubes con más integrantes), indicando cuántos fans tiene. NOTA: se permite usar TOP en la sentencia.

SELECT nombre, num

FROM club

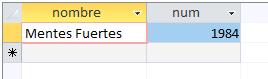
WHERE num IN (SELECT MIN(num)

FROM (SELECT TOP 6 NUM

FROM club

GROUP BY num

ORDER BY num DESC));



33. Obtener el nombre de los artistas que tengan la función de bajo en un único grupo y que además éste tenga más de dos miembros.

SELECT DISTINCT nombre

FROM artista, pertenece

WHERE artista.dni IN (SELECT dni

FROM pertenece

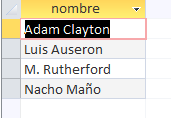
WHERE funcion LIKE 'bajo') AND

cod IN (SELECT cod

FROM pertenece

GROUP BY cod

HAVING COUNT(cod)>1);



34. Obtener el disco con menor duración total, indicando cuántos minutos dura.

SELECT MIN(suma) as Disco\_Menor\_Duracion

FROM (SELECT SUM(cancion.duracion) AS suma

FROM cancion, esta, disco

WHERE disco.cod = esta.cod AND esta.can = cancion.cod

GROUP BY disco.cod);

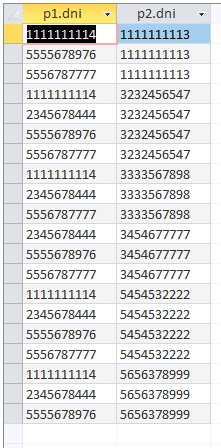


35. Para formar nuevos grupos, se pretende crear dúos formados por un vocalista y un guitarrista que no hayan estado en el mismo grupo anteriormente. Generar todos los grupos posibles.

SELECT p1.dni, p2.dni

FROM pertenece p1, pertenece p2

WHERE (p1.funcion LIKE 'guitarra') AND (p2.funcion LIKE 'voz') AND (p1.cod<>p2.cod)



# TAREAS DE ADMINISTRACIÓN

El software utilizado en esta práctica es Acces 2013 bajo la licencia Microsoft Office Professional Plus 2013 y el sistema operativo Windows 10 Pro bajo licencia SOFTWARE DE MICROSOFT.

# REFERENCIAS BIBLIOGRÁFICAS

* Curso de SQL en línea, Curso SQL - Curso SQL desde cero. (2020). Retrieved 20 April 2020: <http://deletesql.com/viewforum.php?f=5&sid=a7fc565fd60d60b65e5f29826072a039>
* Elmasri, R., Navathe, S. and Escalona García, R., 1997. Sistemas De Bases De Datos. Wilmington: Addison-Wesley.
* Mota Herranz, L., Celma Giménez, M. and Casamayor Ródenas, J., 1994. Bases De Datos Relacionales. Valencia: Universidad Politécnica, Servicio de Publicaciones.