



Instituto Politécnico Nacional
Escuela Superior de Cómputo



Bases de Datos

Práctica no. 3:

Consultas de dos o más tablas

Profesor: Euler Hernández Contreras

Alumno: Calva Hernández José Manuel

Grupo: 2CM12

Índice

Índice.....2

Marco Teórico3

Instrucciones5

Screenshots9

Conclusiones12

Referencias.....12

Marco Teórico

La sentencia SELECT soporta la recuperación de datos de una o más tablas. Esta sección describe un formato simplificado de la sentencia SELECT. La sentencia SELECT descrita aquí tiene el siguiente formato:

```
SELECT <list of columns and expressions usually involving columns>

      FROM <list of tables and join operations>

      WHERE <list of row conditions connected by AND, OR, NOT>

      GROUP BY <list of grouping columns>

      HAVING <list of group conditions connected by AND, OR, NOT>

      ORDER BY <list of sorting specifications>
```

En SQL, los operadores básicos para comparar lógicamente los valores de los atributos entre sí y con constantes literales son =, <, <=, >, >= y <>, que se corresponden con los operadores =, <, >, <=, >= y <> del álgebra relacional, respectivamente, y con los operadores =, <, <=, >, >= y != del lenguaje de programación C/C++.

Consulta 1. Recuperar el nombre y la dirección de todos los empleados que trabajan en el departamento 'Investigación'.

```
C1: SELECT Nombre, Apellido1, Dirección
      FROM EMPLEADO, DEPARTAMENTO
      WHERE NombreDpto='Investigación AND NumeroDpto=Dno;
```

La consulta es parecida a una secuencia SELECCIÓN-PROYECCIÓN-CONCATENACIÓN de operaciones del álgebra relacional. Dichas consultas se denominan a veces consultas selección-proyección-concatenación. En la cláusula WHERE de el, la condición NombreDpto='Investigación' es una condición de selección y se corresponde con una operación SELECCIÓN del álgebra relacional. La condición NumeroDpto=Dno es una condición de concatenación, que corresponde a una condición CONCATENACIÓN del álgebra relacional.

En SQL el mismo nombre se puede utilizar para dos (o más) atributos, siempre y cuando los atributos se encuentren en relaciones diferentes. Si es el caso, y una consulta se refiere a dos o más atributos que tienen el mismo nombre, debemos calificar el nombre del atributo con el nombre de la relación a fin de evitar la ambigüedad. Esto se consigue colocando como prefijo el nombre de la relación al nombre del atributo, y separando los dos nombres con un punto.

La ambigüedad también aparece en el caso de las consultas que se refieren dos veces a la misma relación, como en el siguiente ejemplo:

```
SELECT E.Nombre, E.Apellido01, S.Nombre, S.Apellido01
      FROM EMPLEADO AS E, EMPLEADO AS S
      WHERE E.SuperDni=S.Dni;
```

En este caso, nos permite declarar nombres de relación alternativos, E y S, denominados alias o variables de tupla, para la relación EMPLEADO. Un alias puede seguir a la palabra clave AS, como se muestra en el ejemplo anterior, o puede seguir directamente al nombre de la relación (por ejemplo, escribiendo EMPLEADO E, EMPLEADO S en la cláusula FROM de CS). También es posible renombrar los atributos de la relación dentro de la consulta SQL, asignándoles unos alias. Por ejemplo, si escribimos:

```
EMPLEADO AS E(Np, A1, A2, Dni, Fn, Dir, Sex, Sal, Sdni, Dno)
```

en la cláusula FROM, Np será el alias de Nombre, A 1 de Apellido01, A2 de Apellido02, etcétera.

La coincidencia inexacta mantiene las similitudes que coinciden en algún patrón en lugar de coincidir en una cadena de caracteres idénticos. Uno de los tipos más comunes de coincidencia inexacta es encontrar valores que tengan un prefijo común, tal como “IS4” (nivel 400 de cursos de IS). Utilizar el operador LIKE en conjunto con un patrón de caracteres de coincidencia * para realizar la coincidencia de prefijos. La cadena de caracteres constante ‘IS4%’ significa que la cadena de coincidencia empieza con “IS4” y termina con cualquier cosa. El carácter comodín % coincide con cualquier cadena de caracteres. Tome en cuenta que los caracteres constantes deben encerrarse entre comillas.

Otro tipo común de coincidencia inexacta es la combinación de cadenas que contienen subcadenas. Para realizar este tipo de búsqueda se debe usar un carácter comodín antes y después de la subcadena. Por ejemplo, para encontrar los cursos que contienen la palabra DATABASE en cualquier parte de la descripción del curso, escriba la condición: CrsDesc LIKE ‘*DATABASE*’ en Access. En Oracle sería CrsDesc LIKE ‘%DATABASE%’. El carácter comodín no es el único de búsqueda de patrones. SQL:2003 especifica el guión bajo _ para búsquedas de cualquier carácter. Algunos DBMS, como Access, usan el signo de interrogación ? para la búsqueda de cualquier carácter. Además, la mayoría de los DBMS contienen caracteres de búsqueda de patrones para buscar un rango de caracteres (por ejemplo, los dígitos del 0 al 9) y cualquier carácter de la lista de caracteres. Los símbolos utilizados para estos patrones de búsqueda de caracteres no son estándares. Para ser un experto en las condiciones búsquedas de patrones inexactas, deberá estudiar los caracteres de búsqueda de patrones disponibles en su DBMS.

Además, para realizar la búsqueda de patrones con cadenas de caracteres, puede utilizar búsqueda exacta con el operador de comparación de igualdad =. Por ejemplo, la condición CourseNo = ‘IS480’ coincide con una fila de la tabla Course. Para los patrones de búsquedas exactas e inexactas son importantes las mayúsculas y las minúsculas.

El operador BETWEEN-AND es un operador para comparar una columna numérica o de tipo fecha contra un rango de valores. El operador BETWEEN-AND regresa verdadero si la columna es mayor o igual que el primer valor y menor o igual que el segundo valor.

Instrucciones

1.- Cargar home.sql

```
create database home;  
use home;  
source ...
```

2.- Resolver las siguientes consultas

a) Mostrar el nombre del asociado y la sucursal donde está asignado

```
Select a.nombre, h.nombre  
from asociado a, homedepot h  
where a.homedepot_idHD=h.idhd  
order by 2,1;
```

b) Mostrar el nombre de los clubes existentes en el estado ... DF (CDMX)

```
Select nombre  
from homedepot  
where estado="DF"  
order by 1;
```

```
Select nombre  
from homedepot  
where estado like "Nuev%"  
order by 1;
```

```
Select nombre  
from homedepot  
where estado like "Tamau%"  
order by 1;
```

c) Mostrar el nombre del socio y la tarjeta asignada

```
Select s.nombre, t.nombre  
from socio s, tarjeta t  
where s.idsocio=t.socio_idsocio  
order by 2,1;
```

```
Select s.nombre, t.nombre  
from socio s, tarjeta t  
where s.idsocio=t.socio_idsocio  
and (s.nombre like "Euler%"  
or s.nombre like "% Euler%")  
order by 2,1;
```

d) Mostrar el nombre de los departamentos de una sucursal ... Coacalco

```
Select h.nombre, d.nombre
from homedepot h, depto d, hddepto x
where h.idhd=x.homedepot_idhd
and x.depto_iddepto=d.iddepto
and h.nombre like "Coacalco%"
order by 1;
```

e) Mostrar el crédito que tienen los socios que se apellidan como Garcia...

```
Select nombre, credito
from socio
where (nombre like "Garc%"
or nombre like "%Garc%")
order by 2;
```

f) Cuántos socios tienen la tarjeta DECORE

```
Select COUNT(*)
from socio s, tarjeta t
where s.idsocio=t.socio_idsocio
and t.nombre like "DECOR%";
```

g) Cómo se llaman los socios de la consulta anterior

```
Select s.nombre, t.nombre
from socio s, tarjeta t
where s.idsocio=t.socio_idsocio
and t.nombre like "DECOR%";
```

h) Cuántos asociados tienen como apellido paterno Pérez

```
Select count(*)
from asociado
where nombre like "Pere%"
order by nombre;
```

i) Cómo se llama los asociados de la consulta anterior

```
Select nombre
from asociado
where nombre like "Pere%"
order by nombre;
```

j) Proyectar las sucursales donde están asignados los asociados

```
Select homedepot_idhd
from asociado;
```

//falta proyección

// 1a forma - lee todo, discrimina y ordena; son dos operaciones

```
Select DISTINCT homedepot_idhd
from asociado
order by 1;
```

// 2a forma - accede directamente el index ¡¡más rápido!!

```
select homedepot_idhd
from asociado
group by homedepot_idhd;
```

3.- Mostrar el nombre de las sucursales y sus asociados, de aquellas ubicadas en las siguientes cp 78045, 89360, 25279

```
Select h.nombre, a.nombre
from homedepot h, asociado a
where h.idhd=a.homedepot_idhd
and (h.direccion like "%78045%"
or h.direccion like "%89360%"
or h.direccion like "%25279%")
order by 1,2;
```

4.- Como se llaman los asociados que tiene la sucursal Metepec

```
Select a.nombre
from asociado a, homedepot h
where a.homedepot_idhd=h.idhd
and h.nombre like "Metep%"
order by 1;
```

```
Select a.nombre
from asociado a, homedepot h
where a.homedepot_idhd=h.idhd
and h.nombre like "Tiju%"
order by 1;
```

```
Select a.nombre
from asociado a, homedepot h
where a.homedepot_idhd=h.idhd
and h.nombre like "Tor%"
order by 1;
```

5.- Qué teléfono tienen los socios que tienen la tarjeta Pagos fijos y mostrar la sucursal donde se encuentran dados de alta

```
Select s.nombre, s.tel
from socio s, tarjeta t, hdsocio x, homedepot h
where s.idsocio=t.socio_idsocio
and s.idsocio=x.socio_idsocio
and x.homedepot_idhd=h.idhd
and t.nombre like "Pag% fij%"
order by 1;
```

6.- Mostrar los departamentos que tiene la sucursal tijuana

```
Select h.nombre as "Tienda", d.nombre as "Departamento"
from homedepot h, depto d, hddepto x
where h.idhd=x.homedepot_idhd
and x.depto_iddepto=d.iddepto
and h.nombre like "Tiju%"
order by 2 desc;
```

7.- Como se llaman las sucursales que tienen el depto de pisos

```
Select h.nombre as "Tienda", h.estado
from homedepot h, depto d, hddepto x
where h.idhd=x.homedepot_idhd
and x.depto_iddepto=d.iddepto
and d.nombre like "Pis%"
order by 1;
```

```
Select h.nombre as "Tienda", h.estado
from homedepot h, depto d, hddepto x
where h.idhd=x.homedepot_idhd
and x.depto_iddepto=d.iddepto
and d.nombre like "Tech%"
order by 1;
```


Screenshots

```
mysql> create database home;
Query OK, 1 row affected (0.02 sec)

mysql> use home;
Database changed
mysql> source C:\Users\manue\Documents\Escuela\3er Semestre\Bases de Datos\Scripts\home.sql
Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

```
mysql> Select nombre
-> from homedepot
-> where estado="DF"
-> order by 1;
```

nombre
Centro
Coapa
Coapa_2
Iztapalapa
Mixcoac
San Jeronimo
Tlatilco

7 rows in set (0.00 sec)

```
mysql> Select s.nombre, t.nombre
-> from socio s, tarjeta t
-> where s.id socio=t.socio_id socio
-> order by 2,1;
```

nombre	nombre
ANGELES DE LA CRUZ ANDREE MICHEL IRVING	CREDITO
ARMENDARIZ CRUZ OCTAVIO ARTURO	CREDITO
ARROYO MEJIA ISAAC HUSAI	CREDITO
BUENDIA LOPEZ ENRIQUE OMAR	CREDITO
BUSTOS GUERRERO SERGIO MANUEL	CREDITO
CASTILLO GONZALEZ IVAN	CREDITO
CASTREJON SUAREZ LUIS	CREDITO
CAUICH MARTINEZ JORGE IVAN	CREDITO
CORTEZ ORTIZ ARTURO	CREDITO
CRUZ CHAVEZ EDGAR DANIEL	CREDITO
LEON OROZCO DANIEL	CREDITO
LOPEZ MOSQUEDA ARTURO	CREDITO
MEDINA MARTINEZ JOSE GUADALUPE	CREDITO
MONTALVO LEZAMA RICARDO	CREDITO

```
mysql> Select h.nombre, d.nombre
-> from homedepot h, depto d, hddepto x
-> where h.idhd=x.homedepot_idhd
-> and x.depto_iddepto=d.iddepto
-> and h.nombre like "Coacalco%"
-> order by 1;
```

nombre	nombre
Coacalco	DECORACION
Coacalco	PUERTAS Y VENTANAS
Coacalco	ILUMINACION

3 rows in set (0.00 sec)

```
mysql> Select nombre, credito
-> from socio
-> where (nombre like "Garc%"
-> or nombre like "%Garc%")
-> order by 2;
```

nombre	credito
CASTILLO GARCIA ZURIEL JAVIER	3000
GARCIA VAZQUEZ Nanci MARGARITA	3000
ZUNIGA GARCIA GUSTAVO ALBERTO	3000
GARCIA MORA OSVALDO	5000
GARCIA HERNANDEZ OSVALDO	5600
GARCIA MEDRANO ANDRES NOE	5900
GARCIA CERVANTES ANEL	6000
GUILLLEN GARCIA JOSE JAIR	7400
GARCIA VELASCO VICTOR HUGO	8000
GUTIERREZ GARCIA ROBERTO	8000
GARCIA MARTINEZ LUIS ALBERTO	8050
JUAREZ GARCIA VICTOR DANIEL	8500
RESENDIZ GARCIA JOSE MIGUEL	10300
ROJAS GARCIA RAUL OMAR	10450

14 rows in set (0.00 sec)

```
mysql> Select COUNT(*)
-> from socio s, tarjeta t
-> where s.idsocio=t.socio_idsocio
-> and t.nombre like "DECOR%";
```

COUNT(*)
5

1 row in set (0.00 sec)

```
mysql> Select s.nombre, t.nombre
-> from socio s, tarjeta t
-> where s.idsocio=t.socio_idsocio
-> and t.nombre like "DECOR%";
```

s.nombre	t.nombre
ANGELES DE LA CRUZ ANDREE MICHEL IRVING	DECORE
BUSTOS GUERRERO SERGIO MANUEL	DECORE
CAMACHO MONTOYA RICARDO	DECORE
TORRES RIVERA LOURDES MARICELA	DECORE
ZUÑIGA GARCIA GUSTAVO ALBERTO	DECORE

5 rows in set (0.00 sec)

```
mysql> Select count(*)
-> from asociado
-> where nombre like "Pere%"
-> order by nombre;
```

count(*)
2

1 row in set (0.00 sec)

```
mysql> Select nombre
-> from asociado
-> where nombre like "Pere%"
-> order by nombre;
```

nombre
PEREZ CASAS THELMA DANIELA
PEREZ ROJAS JESUS ENRIQUE

2 rows in set (0.00 sec)

```
mysql> Select DISTINCT homedepot_idhd
-> from asociado
-> order by 1;
```

homedepot_idhd
HD001
HD002
HD003
HD004
HD005
HD006
HD007
HD008
HD009
HD010
HD011
HD012
HD013
HD014
HD015
HD016
HD017
HD018
HD019
HD020
HD021
HD022
HD023
HD024
HD025
HD026
HD027
HD028
HD029
HD030
HD031
HD032
HD033
HD034

34 rows in set (0.00 sec)

```
mysql> select homedepot_idhd
-> from asociado
-> group by homedepot_idhd;
```

homedepot_idhd
HD001
HD002
HD003
HD004
HD005
HD006
HD007
HD008
HD009
HD010
HD011
HD012
HD013
HD014
HD015
HD016
HD017
HD018
HD019
HD020
HD021
HD022
HD023
HD024
HD025
HD026
HD027
HD028
HD029
HD030
HD031
HD032
HD033
HD034

34 rows in set (0.00 sec)

```
mysql> Select h.nombre, a.nombre
-> from homedepot h, asociado a
-> where h.idhd=a.homedepot_idhd
-> and (h.direccion like "%78045%"
-> or h.direccion like "%89360%"
-> or h.direccion like "%25279%")
-> order by 1,2;
```

nombre	nombre
Saltillo	LOBATO RUIZ GABRIEL
Tlalnepantla	LAREDO RAZA DAVID

2 rows in set (0.00 sec)

```
mysql> Select a.nombre
-> from asociado a, homedepot h
-> where a.homedepot_idhd=h.idhd
-> and h.nombre like "Metep%"
-> order by 1;
```

nombre
RAMIREZ BARQUIN MIGUEL ANGEL

1 row in set (0.00 sec)

```
mysql> Select s.nombre, s.tel
-> from socio s, tarjeta t, hdsocio x, homedepot h
-> where s.idsocio=t.socio_idsocio
-> and s.idsocio=x.socio_idsocio
-> and x.homedepot_idhd=h.idhd
-> and t.nombre like "Pag% fij%"
-> order by 1;
```

nombre	tel
ANGELES DE LA CRUZ ANDREE MICHEL IRVING	01(55)51-22-33-00
AÑORVE MORALES ANTONIO CESAR	01(55)51-22-33-00
ARMENDARIZ CRUZ OCTAVIO ARTURO	01(55)51-22-33-00
CAMACHO MONTOYA RICARDO	01(55)51-22-33-00
CARACHEO EVANGELISTA IVAN	01(55)51-22-33-00
CARBAJAL NUÑEZ NOE DANIEL	01(55)51-22-33-00
CARBAJAL NUÑEZ NOE DANIEL	01(55)51-22-33-00
CARVAJAL MURILLO ANA VICTORIA	01(55)51-22-33-00
CARVAJAL MURILLO ANA VICTORIA	01(55)51-22-33-00
CASTAÑEDA GONZALEZ LEONARDO MAURICIO	01(55)51-22-33-00
CASTILLO GARCIA ZURIEL JAVIER	01(55)51-22-33-00
CASTILLO GONZALEZ IVAN	01(55)51-22-33-00
GARCIA VAZQUEZ NANCI MARGARITA	01(55)51-22-33-00
GARCIA VELASCO VICTOR HUGO	01(55)51-22-33-00
GARDUÑO SANCHEZ LUIS ALBERTO	01(55)51-22-33-00
GASTELUM SARO VICTOR MANUEL	01(55)51-22-33-00
GASTELUM SARO VICTOR MANUEL	01(55)51-22-33-00
GUTIERREZ RESENDIZ ROBERTO	01(55)51-22-33-00
GUZMAN RIOS RAFAEL ANGEL	01(55)51-22-33-00
HERNANDEZ ROSAS DANIEL ALEJANDRO	01(55)51-22-33-00
HERRERA AGUILAR ALEJANDRO DANIEL	01(55)51-22-33-00
LAGOS CAXNAJOY EDGAR JACOBO	01(55)51-22-33-00

22 rows in set (0.02 sec)

```
mysql> Select h.nombre as "Tienda", d.nombre as "Departamento"
-> from homedepot h, depto d, hddepto x
-> where h.idhd=x.homedepot_idhd
-> and x.depto_iddepto=d.iddepto
-> and h.nombre like "Tiju%"
-> order by 2 desc;
```

Tienda	Departamento
Tijuana	TECHOS
Tijuana	PLOMERIA
Tijuana	ILUMINACION

3 rows in set (0.00 sec)

```
mysql> Select h.nombre as "Tienda", h.estado
-> from homedepot h, depto d, hddepto x
-> where h.idhd=x.homedepot_idhd
-> and x.depto_iddepto=d.iddepto
-> and d.nombre like "Pis%"
-> order by 1;
```

Tienda	estado
Durango	Durango
Ensenada	Baja California
Henequen	Chihuahua
León Centro Max	Guanajuato
Puebla Norte	Puebla
Queretaro	Queretaro
San Nicolás/Nogalar	Nuevo Leon

7 rows in set (0.00 sec)

Conclusiones

Con esta práctica comenzamos las consultas de una manera más formal respecto de la breve introducción realizada en la práctica anterior, además, iniciamos las consultas con más de una tabla en su haber, aprendiendo la forma de realizar la una correcta formulación de esta para poder interrelacionar las tablas de una manera óptima.

A su vez, comenzamos a aplicar restricciones más complejas en las consultas a manera de refinar la búsqueda de manera más particular para nuestro interés, o bien, lo solicitado por el profesor. Para ello, usamos operador como AND, OR, BETWEEN o LIKE; cada uno de ellos con un propósito diferente.

Lo más complicado de la práctica es hacer una correcta relación de las tablas, ya que si no se hace de una manera óptima puede resultar en una pérdida de tiempo de procesamiento, o peor aún, si no se realiza de manera correcta, nuestra consulta puede arrojar resultados nulos o erróneos, lo cual no es de interés para nosotros.

Referencias

- Ramez, E., & Navathe, S. (2000). *Sistemas de Bases de Datos: Conceptos Fundamentales* (1st ed.). México: Pearson Educación.
- Ricardo, C., Campos Olguín, V., & Enríquez Brito, J. (2010). *Bases de datos*. México: McGraw-Hill Interamericana.
- Mannino, M. (2007). *Administración de Bases de Datos. Diseño y desarrollo de aplicaciones* (3rd ed.). México: McGraw-Hill Interamericana.