



UNIVERSIDAD TECNOLÓGICA DE PUEBLA

**TECNOLOGÍAS DE LA INFORMACIÓN Y LA
COMUNICACIÓN (SISTEMAS INFORMATICOS)**

MATERIA: BASE DE DATOS II

PROFESOR: JOSÉ FRANCISCO ESPÍNOSA GARITA

PRODUCTO N° 2

INTEGRANTES:

JUAN LUIS SANTAMARIA VAZQUEZ

MARTÍN ARIAS CEBADA

3° "H" TURNO VESPERTINO

CUATRIMESTRE MAYO-AGOSTO 2017

ÍNDICE

INTRODUCCIÓN	3
DEFINICIONES	4
TABLAS.....	5
CONSULTAS JOIN	7
.....	8
SUBCONSULTAS	8
CONSULTAS CON OPERACIONES DE CONJUNTOS	10
CONSULTAS CON PRODUCTO CARTESIANO.....	11
JOIN	12
PRODUCTO CARTESIANO	12
CONSULTA LOCAL	12
CONSULTA REMOTA	12
EXPLICACIONES.....	13
CONCLUSIÓN	14
BIBLIOGRAFÍA	14

INTRODUCCIÓN

En el presente trabajo se hablara sobre la realización de consultas, esto se lleva gracias a la base de datos que se realizó con anterioridad, metiendo cada uno de los campos en cada una de sus tablas como lo que fueron de cliente, administrador, proveedor y así metiendo valores correspondientes a esas tablas. De esa forma se podrán realizar las consultas de una, dos tablas o en las que se desean realizar las consultas necesarias gracias a esto poder cumplir un buen requerimiento de datos. Las consultas sirven para visualizar los datos que tienen las tablas, de esta forma poder visualizar mejor la relación que poseen, otra parte la unión de estas hace que veamos la composición de sus atributos entre las tablas y que datos están en esa tabla.

Una consulta sirve para poder buscar dentro de todos tus datos solo los que desees encontrar, para realizar sumas, restas, multiplicaciones con lo operadores aritméticos, puedes contar cuantos registros tienes y en dado caso de que algunas tablas no concuerden arreglar esos problemas como se vio en el producto anterior, con uso del comando `alter table ... constraint`. Las consultas son de gran utilizada si las sabes manejar y tienes experiencia con ellas, solo debes razonar un poco y darle lógica.

DEFINICIONES

Consultas avanzadas: Las consultas mutitabla son llamadas así porque están basadas en más de una tabla. Otra manera de llamarlas es como Combinaciones. Las combinaciones permiten recuperar datos de dos o más tablas según las relaciones lógicas entre ellas. Las combinaciones indican cómo debe usar Microsoft SQL Server los datos de una tabla para seleccionar las filas de otra tabla. Una condición de combinación define la forma en la que dos tablas se relacionan en una consulta al:

- Especificar la columna de cada tabla que debe usarse para la combinación. Una condición de combinación típica especifica una clave externa de una tabla y su clave asociada en otra tabla.
- Especificar un operador lógico (por ejemplo, = o <>) para usarlo en los valores de comparación de las columnas.

Las combinaciones internas se pueden especificar en las cláusulas FROM o WHERE. Las combinaciones externas sólo se pueden especificar en la cláusula FROM. Las condiciones de combinación se combinan con las condiciones de búsqueda de WHERE y HAVING para controlar cuáles son las filas seleccionadas de las tablas base a las que se hace referencia en la cláusula FROM.

Descripción de SGBD SQL Server: El sistema SQL Server es un sistema gestor de base de datos con el cual puedes interactuar, esto permite la interacción de casi cualquier usuario, debido a que te ayuda con la corrección de errores y tienes dos formas de crear una base de datos, por medio de scrip o creando tablas como lo harías en Word (solo rellenando los campos). Es muy útil aunque un poco distinto de MariaDB con la sintaxis.

TABLAS

Tabla Proveedor

	id	Nombre	ApellidoP	ApellidoM	Calle	NoEx	NoIn	Colonia	Municipio	CodigoP	Telefono
1	1	Pascual	Meneces	Montes	la Cañada	65	78	sanBuena	Papalotla	87487	2224214587
2	2	JuanLuis	Santamaria	Vazquez	la calderon	52	87	Panazacola	Zacatelco	90796	2224214887
3	3	Luis	Vargas	Hemadex	del trabajo	92	84	Buena Vista	San Totonbio	87487	2222458761
4	4	Sergio	Salgado	Agulla	la esperanza	72	44	Sabta cruz	Papalotla	1254	2222231475
5	5	Aldo	Cortez	Flores	las Rosas	32	98	san Lorenzo	Papalotla	7841	2222478987

Tabla Sucursales

	id	Nombre	Calle	NoEx	NoIn	Colonia	Municipio	CodigoP	Telefono	CorreoElectronico	Facebook
1	1	Alfredo	carranza	98	97	los pinos	nativitas	78945	2222879851	black87@gmail.com	alfredo Gevara
2	2	Roberto	la agulla	28	37	La Discordia		78974	2222884541	bdgd@gmail.com	la canatada
3	3	Luis	el nopal	87	38	La Loma del Blanca	San Juan	78945	5422879541	bladhudh7@gmail.com	algien que no ere tu
4	4	Eliot	casi illegas	47	46	caracol	capulin	77845	2222879541	ehuw7@gmail.com	alfregdy
5	5	Vicente	la mar	74	87	los cabos	Baranca Onda	98945	2872879541	fhewru@gmail.com	conejoblaack

Tabla ProveedorSucursales

	idP	idS
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

Tabla Administrador

	idAdministrador	idS	Nombre	ApellidoP	ApellidoM	Usuario	Contraseña	Permisos
1	1	1	Edgar	Mesa	Flores	Edgarelmaso445	754414454	ofertas
2	2	2	Antonio	Duran	Montes	Antoniocama	papu	diseño
3	3	3	Margarita	Quiros	Contreras	Margarita445	v7644	precio
4	4	4	Pablo	Mesa	Sanches	SANCHES	ulfihe474564	portada
5	5	5	Escutia	Lima	Calderon	Calderono445	rg445445	pedidos

Tabla Producto

	idProducto	Nombre	Precio	Tipo	Relleno	Sabor	Porciones	Tamaño	GelatinaDe	DiseñoPastel
1	1	pan	6	1	NULL	NULL	NULL	NULL	NULL	NULL
2	2	Gelatina	17	2	NULL	NULL	NULL	NULL	NULL	NULL
3	3	Postre	24	3	NULL	NULL	NULL	NULL	NULL	NULL
4	4	Choco fresa	400	4	fresa	chocolate	40	NULL	NULL	Plataforma
5	5	pan	12	1	NULL	NULL	NULL	NULL	NULL	NULL

Tabla SucursalesProducto

	idS	idProducto
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5

Tabla Cotizacion

	idCotizacion	precio	descripcion
1	1	8000	NULL
2	2	3000	NULL
3	3	5000	NULL
4	4	6000	NULL
5	5	7000	NULL

Tabla ProductoCotizacion

	idCotizacion	precio	descripcion
1	1	8000	NULL
2	2	3000	NULL
3	3	5000	NULL
4	4	6000	NULL
5	5	7000	NULL

Tabla Cliente

	idCliente	Nombre	ApellidoP	ApellidoM	contraseña	Calle	NoEx	NoIn	Colonia	Municipio	CodigoP	Telefondo
1	1	Genaro	Ramirez	Carranza	contrasena	albaro obregon	55	2	Villa frontera	San Francisco	80365	222898
2	2	Jorge	Corona	Zayas	12345	buganblias	33	NULL	fraccionamiento 5°	la resureccion	20014	222898
3	3	Javier	Vargas	Zapata	lolli	la cienega	45	NULL	California	san felipe	44478	222878
4	4	Gustavo	Leon	Lara	joescriuz	xaltipa	2	2	via corta	san Ramon	66365	222841
5	5	Genaro	Tamariz	Lala	alpura	solitaria	15	7	galaxia	Amalucan	80355	229898

Tabla Pedidos

	id	idCliente	Precio	Apartado	FechaEntrega	FechaEmision	Calle	NoEx	NoIn	Colonia	Municipio	CodigoP	CantidadPro
1	1	1	800	1	2017-06-01	2017-06-01	Altamirano	125	NULL	Amalucan	Puebla	72894	2
2	2	2	500	0	2017-06-14	2017-06-03	Benito Juarez	99	NULL	Buganblias	Puebla	72874	1
3	3	3	1000	1	2017-06-15	2017-06-10	3 sur	1245	NULL	Centro	Puebla	72390	1
4	4	4	200	0	2017-06-06	2017-06-01	13 norte	64	NULL	San Diego	Puebla	72300	2
5	5	5	100	1	2017-06-08	2017-06-11	2 sur	48	NULL	Centro	Puebla	72600	1

Tabla Ventas

	idVentas	idP	PrecioVenta	CantidadProductos	Descripcion	NombreProducto
1	1	1	500	2	NULL	Pastel
2	2	2	200	20	NULL	Pan
3	3	3	1000	1	NULL	Pastel
4	4	4	2000	1	NULL	Pastel
5	5	5	800	3	NULL	Bocadillos

CONSULTAS JOIN

--Join, Inner join, left join y right join--

```
select idP, PrecioVenta, CantidadProductos
from Ventas as V inner join Pedidos as P on V.idP=P.id
```

```
select P.idCliente, FechaEmision, FechaEntrega
from Pedidos as P inner join cliente as C on P.idCliente=C.idCliente
```

/*Se crearon dos tablas para realizar las consultas con right join y left join*/

```
select idCliente, Nombre
Into ClienteA
from cliente
Where idCliente between 1 and 3;
```

```
select idCliente, Nombre
Into ClienteB
from cliente
Where idCliente between 2 and 5;
```

```
select *
from ClienteA CA left join ClienteB CB on CA.idCliente=CB.idCliente
```

```
select *
from ClienteA CA left join Pedidos P on CA.idCliente=P.idCliente
```

```
select *
from ClienteA CA right join ClienteB CB on CA.idCliente=CB.idCliente
```

idP		PrecioVenta	CantidadProductos
1	1	500	2
2	2	200	20
3	3	1000	1
4	4	2000	1
5	5	800	3

idCliente		FechaEmision	FechaEntrega
1	1	2017-06-01	2017-06-01
2	2	2017-06-03	2017-06-14
3	3	2017-06-10	2017-06-15
4	4	2017-06-01	2017-06-06
5	5	2017-06-11	2017-06-08

idCliente		Nombre	idCliente		Nombre
1	1	Genaro	NULL	NULL	
2	2	Jorge	2	Jorge	
3	3	Javier	3	Javier	

idCliente		Nombre	id	idCliente		Precio	Apartado	FechaEntrega	FechaEmision	Calle	NoEx	NoIn	Colonia	Municipio	Codig
1	1	Genaro	1	1		800	1	2017-06-01	2017-06-01	Altamirano	125	NULL	Amalucan	Puebla	7289
2	2	Jorge	2	2		500	0	2017-06-14	2017-06-03	Benito J...	99	NULL	Buganbi...	Puebla	7287
3	3	Javier	3	3		1000	1	2017-06-15	2017-06-10	3 sur	1245	NULL	Centro	Puebla	7239

idCliente		Nombre	idCliente		Nombre
1	2	Jorge	2	Jorge	
2	3	Javier	3	Javier	
3	NULL	NULL	4	Gusta...	
4	NULL	NULL	5	Genaro	

SUBCONSULTAS

--Subconsultas--

```
select idCliente, Nombre, ApellidoP, ApellidoM
from cliente
where NoEx>10 and NoEx<100 and idCliente in (select idCliente from Pedidos where
Municipio='Puebla')
```

```
select idVentas, PrecioVenta, CantidadProductos
from Ventas
where PrecioVenta>300 and PrecioVenta<1000 and idP in (select id from Pedidos where
Municipio='Puebla')
```

```
select idAdministrador, Nombre, ApellidoP, Usuario
from Administrador
where idAdministrador>1 and idAdministrador<5 and idS in (select id from Sucursales
where NoEx>10 and NoEx<200)
```

```
select idAdministrador, Nombre, ApellidoP, Usuario
from Administrador
where idAdministrador>1 and idAdministrador<5 and idS in (select id from Sucursales
where NoIn>10 and NoIn<200)
```

```
select idVentas, PrecioVenta, CantidadProductos
from Ventas
where PrecioVenta>300 and PrecioVenta<1000 and idP in (select id from Pedidos where
FechaEmision='2017/06/01')
```


100 % < >				
Resultados Mensajes				
	idCliente	Nombre	ApellidoP	ApellidoM
1	1	Genaro	Ramirez	Carranza
2	2	Jorge	Corona	Zayas
3	3	Javier	Vargas	Zapata
4	5	Genaro	Tamariz	Lala
	idVentas	PrecioVenta	CantidadProductos	
1	1	500	2	
2	5	800	3	
	idAdministrador	Nombre	ApellidoP	Usuario
1	2	Antonio	Duran	Antoniocama
2	3	Margarita	Quiros	Margarita445
3	4	Pablo	Mesa	SANCHES
	idAdministrador	Nombre	ApellidoP	Usuario
1	2	Antonio	Duran	Antoniocama
2	3	Margarita	Quiros	Margarita445
3	4	Pablo	Mesa	SANCHES
	idVentas	PrecioVenta	CantidadProductos	
1	1	500	2	

CONSULTAS CON OPERACIONES DE CONJUNTOS

Se crearon dos tablas para poder hacer las operaciones de conjuntos.

```
/*Se crearon dos tablas para realizar las consultas con right join and left join*/
```

```
select idCliente, Nombre  
Into ClienteA  
from cliente  
Where idCliente between 1 and 3;
```

```
select idCliente, Nombre  
Into ClienteB  
from cliente  
Where idCliente between 2 and 5;
```

```
--Union, interseccion y complemento--
```

```
select *  
from ClienteA  
union  
select *  
from ClienteB
```

```
select *  
from ClienteA  
intersect  
select *  
from ClienteB
```

```
select *  
from ClienteA  
except  
select *  
from ClienteB
```

```
select *  
from ClienteB  
except  
select *  
from ClienteA
```

```
select *  
from ClienteB  
union  
select *  
from ClienteA
```

Resultados		Mensajes
	idCliente	Nombre
1	1	Genaro
2	2	Jorge
3	3	Javier
4	4	Gusta...
5	5	Genaro

	idCliente	Nombre
1	2	Jorge
2	3	Javier

	idCliente	Nombre
1	1	Genaro

	idCliente	Nombre
1	4	Gustavo
2	5	Genaro

	idCliente	Nombre
1	1	Genaro
2	2	Jorge
3	3	Javier
4	4	Gusta...
5	5	Genaro

CONSULTAS CON PRODUCTO CARTESIANO

--Producto Cartesiano--

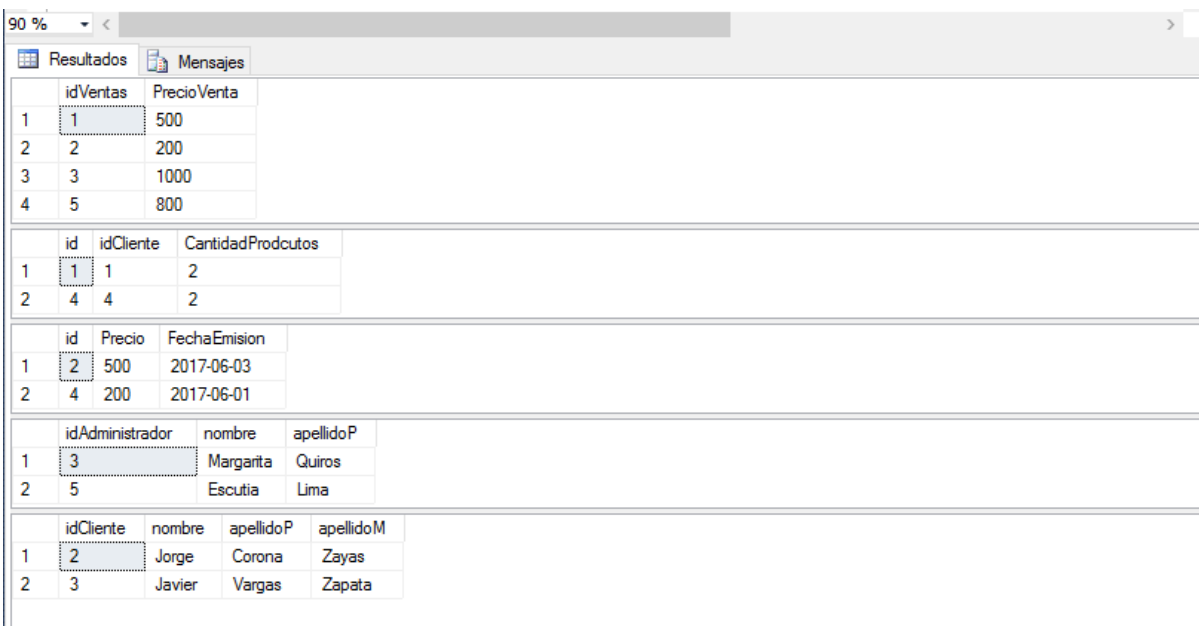
```
select idVentas, PrecioVenta from Ventas where PrecioVenta>10 and PrecioVenta<2000
```

```
select id, idCliente, CantidadProdutos from Pedidos where CantidadProdutos>1 and CantidadProdutos<5
```

```
select id, Precio, FechaEmision from Pedidos where Precio>100 and Precio<800
```

```
select idAdministrador, nombre, apellidoP from Administrador where Nombre like '%a'
```

```
select idCliente, nombre, apellidoP, apellidoM from cliente where ApellidoM like 'z%'
```



The screenshot shows a database query results window with a zoom level of 90%. It displays five tables of data, each with a grid icon and a 'Mensajes' button. The tables are as follows:

	idVentas	PrecioVenta
1	1	500
2	2	200
3	3	1000
4	5	800

	id	idCliente	CantidadProdutos
1	1	1	2
2	4	4	2

	id	Precio	FechaEmision
1	2	500	2017-06-03
2	4	200	2017-06-01

	idAdministrador	nombre	apellidoP
1	3	Margarita	Quiros
2	5	Escutia	Lima

	idCliente	nombre	apellidoP	apellidoM
1	2	Jorge	Corona	Zayas
2	3	Javier	Vargas	Zapata

JOIN

Las condiciones de la combinación en la cláusula FROM ayuda a separarlas de cualquier otra condición de búsqueda que se pueda especificar en una cláusula WHERE; es el método recomendado para especificar combinaciones. La sintaxis simplificada de la combinación de la cláusula FROM es:

```
FROM first_table join_type second_table [ON (join_condition)]
```

join_type especifica el tipo de combinación que se lleva a cabo: interior, exterior o cruzada. join_condition define el predicado que se va a evaluar en cada par de filas combinadas.

PRODUCTO CARTESIANO

El producto cartesiano es un tipo de composición de tablas, aplicando el producto cartesiano a dos tablas se obtiene una tabla con las columnas de la primera tabla unidas a las columnas de la segunda tabla, y las filas de la tabla resultante son todas las posibles concatenaciones de filas de la primera tabla con filas de la segunda tabla. El producto cartesiano se indica poniendo en la FROM las tablas que queremos componer separadas por comas, podemos obtener así el producto cartesiano de dos, tres, o más tablas.

CONSULTA LOCAL

La consulta local es aquella que se efectúa en el equipo donde se encuentra almacenada la base de datos, utilizando como herramienta principal MySQL server, creándose una base de datos, con sus tablas y atributos, las cuales como forma más sencilla es por medio de scripts, de esta forma para poder hacer consultas o modificaciones solo se necesita hacer un new query para poder acceder a la base de datos preexistente, ya que MySQL fue creado en principio para uso restringido de los datos y seguridad de la información.

No obstante existen formas de poder acceder a esta información por medio de otro equipo ajeno a nuestra base de datos.

CONSULTA REMOTA

Por norma general la mayoría de servidores tan solo permiten que las conexiones a MySQL se realicen en local, es decir, desde el mismo servidor Cloud Hosting, y no desde un servidor o equipo externo. En los “setups” en los que es necesario configurar MySQL en un servidor aparte, para no sobrecargar el frontal web por

ejemplo, se tienen que llevar a cabo algunas configuraciones para poder conectar de forma remota a las bases de datos.

La forma correcta de hacerlo es:

- Acceder al servidor a través de SSH como usuario “root”
- Se le debe dar privilegios de acceso remoto a la base de datos que a la que se desea acceder
- Finalmente, lo que se deberá hacer es habilitar el puerto 3306 en las iptables.

EXPLICACIONES

¿Qué diferencia existe entre el producto cartesiano y el join?

El producto cartesiano es una consulta un poco mediocre, debido a que se queda corta de funciones y su única función es crear una nueva tabla relacionando ambas, como si multiplicaras $A*B$, todo lo que está en la tabla de A se multiplica por B, dándote valores que tal vez no buscabas y haciéndote perder el tiempo. Mientras que Join es más eficiente, más precisa y ajusta los valores a pesar de ser diferentes entre ambas tablas, además de incorporar otras funciones como lo son left join, right join y cross join.

Consultas remotas y locales

Existen distintas consultas como se vio anteriormente, pero se clasifican en dos y es muy fácil distinguirlas. Las consultas locales son aquellas que se hacen cuando la base de datos esta almacenada en la memoria cache y tú puedes hacer las consultas que quieras pero es una base de datos completamente estática. Por otro lado con las consultas remotas, la base de datos también se almacena en una memoria cache pero en diferencia con la otra, los usuarios son los que manejan y le dan vida a esta base de datos.

CONCLUSIÓN

Se puede decir que todo comenzó desde el primer producto, con el proyecto que se asignó por el comienzo del curso este proyecto consistía en cubrir todos los aspectos posibles de este hasta donde se pudiera después hacer un análisis para poder realizar lo que es una base con todo lo que se ha registrado con base al proyecto , se empezó por encontrar todas las entidades posibles con sus correspondientes campos luego se crearían los modelos E-R y M-R esto con una normalización para poder reducir todos los campos que pudieran generar alguna redundancia de datos por consiguiente se crearía la base de datos en la cual se insertaran todas las tablas con sus atributos , a su vez se le meterían datos simulando un registro de la base esto para poder realizar todo lo que se pedía en la rúbrica del curso como las consultas inserciones y cada uno de los requerimientos que se pedían.

BIBLIOGRAFÍA

- [Antología de base de datos II proporcionada por el profesor José Francisco Espinosa Garita.](#)
- [Herramientas tecnológicas: SQL Server.](#)