

# Universidad Tecnológica de Puebla

“Tecnologías de la Información y  
Comunicación”

“PRODUCTO 2”

Base de Datos 2

Prof. José Francisco Espinoza Garita

Alumno:  
Romero Tamariz Eder  
Menor de Jesús Samuel

3° H

1994

# ÍNDICE

Introducción.....	3
Definiciones.....	4
Tablas .....	5
Join, Inner join, left join y right join.....	8
Subconsultas.....	10
Operadores de conjuntos.....	12
Producto Cartesiano.....	15
Definiciones&Explicaciones.....	17
Conclusión.....	18

# INTRODUCCIÓN

En este producto se hablará 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 los 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

Es una consulta anidada en una instrucción SELECT, INSERT, UPDATE o DELETE, o bien en otra subconsulta. Las subconsultas se pueden utilizar en cualquier parte en la que se permita una expresión.

## Combinaciones

Conocidas como combinaciones o consultas multitabla, estas son llamadas así ya que están basadas en más de una tabla, permiten recuperar datos de dos o más tablas según las relaciones lógicas existentes entre ellas; indican como debe usar

Microsoft SQL Server los datos de una tabla para seleccionar las filas de otra tabla.

Algunas condiciones de combinación definen la forma en la que dos tablas se relacionan al hacer una consulta al momento de:

- Especificar la columna de cada tabla que debe usarse para la combinación (esta es una condición de combinación típica específica 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.

## Join

Las condiciones en la combinación de la cláusula (FROM) ayudan a separar de cualquier otra condición de búsqueda que se pueda especificar en una cláusula WHERE (este método es el recomendado para especificar combinaciones). La sintaxis simplificada de la combinación de la cláusula FROM sería: 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. Mientras que *join\_condition* define el predicado que se va a evaluar en cada par de filas combinadas.

## SGBD SQL server

SQL Server es un sistema de gestión de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact-SQL), un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas.

# Tablas con registros apropiados

--Datos Direccion--

```
insert into Direccion values(10,1,'Puebla','Cholula','13 Norte',201,'2226994121','72310');
insert into Direccion values(7,2,'Veracruz','Xalapa','5 sur',150,'2225462453','35634');
insert into Direccion values(5,3,'Sonora','Hermosillo','3 norte',500,'22242453435','64524');
insert into Direccion values(8,4,'Tabasco','Villahermosa','3 E',110,'2221456353','14535');
insert into Direccion values(12,5,'Chiapas','Tuxtla Gutierrez','5 Este',145,'2227525332','98345');
```

100 %

Resultados Mensajes

	IdDireccion	IdUsuario	Estado	Ciudad	Calle	Numero	Telefono	CodigoPostal
1	5	3	Sonora	Hemosillo	3 norte	500	22242453435	64524
2	7	2	Veracruz	Xalapa	5 sur	150	2225462453	35634
3	8	4	Tabasco	Villahemosa	3 E	110	2221456353	14535
4	10	1	Puebla	Cholula	13 Norte	201	2226994121	72310
5	12	5	Chiapas	Tuxtla Gutierrez	5 Este	145	2227525332	98345

--Datos Usuario--

```
insert into Usuario values(1,'eder.play1000@gmai.com','edgr463356','Eder','Romero','Tamariz');
insert into Usuario values(2,'edder1000@hotmail.com','ghhtts100','Heder','Manstranso','Lopez');
insert into Usuario values(3,'javigal.10@hotmail.com','et423','Javier','Perez','Zapata');
insert into Usuario values(4,'luis.thor@gmail.com','mngs45','Luis','Santamaria','Vazquez');
insert into Usuario values(5,'packsual12@hotmail.com','asdf24500','Jose','Pascual','Jimenez');
```

100 %

Resultados Mensajes

	IdUsuario	Email	Contraseña	Nombre	ApellidoP	ApellidoM
1	1	eder.play1000@gmai.com	edgr463356	Eder	Romero	Tamariz
2	2	edder1000@hotmail.com	ghhtts100	Heder	Manstranso	Lopez
3	3	javigal.10@hotmail.com	et423	Javier	Perez	Zapata
4	4	luis.thor@gmail.com	mngs45	Luis	Santamaria	Vazquez
5	5	packsual12@hotmail.com	asdf24500	Jose	Pascual	Jimenez

```
--Datos Comics--
```

```
insert into Comics values(8,'Civil War','Mark Millar','Editorial Televisa','Accion',25,'2001');
insert into Comics values(9,'Spiderman','Steve McNiven','Editorial Televisa','Accion',20,'2011');
insert into Comics values(10,'Superman','Brian Michael Bendis','Warner Bros','Accion',22,'1998');
insert into Comics values(11,'The Incredible Hulk','Joe Simon','Editorial Televisa','Accion',21,'1889');
insert into Comics values(12,'Liga de la Justicia','Jack Kirby','Warner Bros','Accion',23,'2005');
```

100 %

Resultados Mensajes

	ISBN	Titulo	Autor	Editorial	Genero	NoComic	FechaDePublicacion
1	8	Civil War	Mark Millar	Editorial Televisa	Accion	25	2001
2	9	Spiderman	Steve McNiven	Editorial Televisa	Accion	20	2011
3	10	Superman	Brian Michael Bendis	Warner Bros	Accion	22	1998
4	11	The Incredible Hulk	Joe Simon	Editorial Televisa	Accion	21	1889
5	12	Liga de la Justicia	Jack Kirby	Warner Bros	Accion	23	2005

```
--Datos Manga--
```

```
insert into Manga values(18,'Bleach','Eiichiro Oda','Panini Editorial','Josei manga',31,'2001','2003');
insert into Manga values(19,'Tokyo Ghoul','Masashi Kishimoto','Nomra Editorial','Seinen manga',32,'1998','2000');
insert into Manga values(20,'Death Note','Karuho Shiina','Ivréa Editorial','Shōjo manga',34,'2005','2008');
insert into Manga values(21,'Parysite','Hiro Mashima','Ecc Ediciones','Gore',33,'2010','2013');
insert into Manga values(22,'Naruto','Takeshi Obata','Selecta Vision','Progresivo',35,'2006','2009');
```

100 %

Resultados Mensajes

	IdManga	Titulo	Autor	Editorial	Genero	NoTomo	FechaDePublicacion	FechaDeFinalizacion
1	18	Bleach	Eiichiro Oda	Panini Editorial	Josei manga	31	2001	2003
2	19	Tokyo Ghoul	Masashi Kishimoto	Nomra Editorial	Seinen manga	32	1998	2000
3	20	Death Note	Karuho Shiina	Ivréa Editorial	Shojo manga	34	2005	2008
4	21	Parysite	Hiro Mashima	Ecc Ediciones	Gore	33	2010	2013
5	22	Naruto	Takeshi Obata	Selecta Vision	Progresivo	35	2006	2009

```
--Datos compra--
```

```
insert into compra values(1,18,8);
insert into compra values(2,19,9);
insert into compra values(3,20,10);
insert into compra values(4,21,11);
insert into compra values(5,22,12);
```

100 %

Resultados Mensajes

	IdUsuario	IdManga	ISBN
1	1	18	8
2	2	19	9
3	3	20	10
4	4	21	11
5	5	22	12

# EJERCICIOS

--Tablas que se generan para los ejercicios Join, Inner join, left join y right join--

## USUARIO A

```
select IdUsuario, Nombre
into UsuarioA
FROM usuario
where IdUsuario between 1 and 2;
select * from UsuarioA;
```

100 % <

	IdUsuario	Nombre
1	1	Eder
2	2	Heder

## Usuario B

```
select IdUsuario, Nombre
into UsuarioB
FROM usuario
where IdUsuario between 2 and 5;
select * from UsuarioB;
```

100 % <

	IdUsuario	Nombre
1	2	Heder
2	3	Javier
3	4	Luis
4	5	Jose

## LEFT JOIN

```
select *  
from UsuarioA EA left join UsuarioB EB on EA.IdUsuario=EB.IdUsuario
```

100 %

Resultados Mensajes

	IdUsuario	Nombre	IdUsuario	Nombre
1	1	Eder	NULL	NULL
2	2	Heder	2	Heder

## Join

```
select*  
from UsuarioA EA full join UsuarioB EB on EA.IdUsuario= Eb.IdUsuario
```

100 %

Resultados Mensajes

	IdUsuario	Nombre	IdUsuario	Nombre
1	1	Eder	NULL	NULL
2	2	Heder	2	Heder
3	NULL	NULL	3	Javier
4	NULL	NULL	4	Luis
5	NULL	NULL	5	Jose

## Right Join

```
select*  
from UsuarioA EA right join UsuarioB EB on EA.IdUsuario= EB.IdUsuario
```

100 %

Resultados Mensajes

	IdUsuario	Nombre	IdUsuario	Nombre
1	1	Eder	NULL	NULL
2	2	Heder	2	Heder
3	NULL	NULL	3	Javier
4	NULL	NULL	4	Luis
5	NULL	NULL	5	Jose



## Inner Join

```
select Email, Contraseña
from Usuario as U inner join usuarioB as C on U.IdUsuario=C.IdUsuario
```

100 %

Resultados Mensajes

	Email	Contraseña
1	edder1000@hotmail.com	ghths100
2	javigal.10@hotmail.com	et423
3	luis.thor@gmail.com	mngs45
4	packsual12@hotmail.com	asdf24500

## Cross Join

```
select *
from UsuarioA EA cross join UsuarioA EB
```

100 %

Resultados Mensajes

	IdUsuario	Nombre	IdUsuario	Nombre
1	1	Eder	1	Eder
2	2	Heder	1	Eder
3	1	Eder	2	Heder
4	2	Heder	2	Heder

## Left Join

```
select *
from UsuarioA EA left join compra P on EA.IdUsuario=P.IdUsuario
```

100 %

Resultados Mensajes

	IdUsuario	Nombre	IdUsuario	IdManga	ISBN
1	1	Eder	1	18	8
2	1	Eder	1	74445	53566
3	2	Heder	2	30	9
4	2	Heder	2	67363	36342

# SUBCONSULTAS

```
select IdManga, NoTomo
from Manga
where NoTomo<100 and IdManga in (select IdManga from Manga where Titulo='Bleach')
```

00 % <

Resultados Mensajes

	IdManga	NoTomo
1	18	31

```
select ISBN, NoComic
from Comics
where NoComic<300 and ISBN in (select ISBN from Comics where Titulo='Civil War')
```

100 % <

Resultados Mensajes

	ISBN	NoComic
1	53566	35
2	8	25

```

select ISBN, NoComic, FechaDePublicacion
from Comics
where NoComic<300 and FechaDePublicacion<3000 and ISBN in (select ISBN from Comics where Editorial='Editorial Televisa')

```

100 %

Resultados Mensajes

	ISBN	NoComic	FechaDePublicacion
1	11	21	1889
2	36342	20	2011
3	53566	35	2001
4	8	25	2001
5	9	20	2011
6	98464	53	1889

```

select ISBN, IdManga, IdUsuario
from compra
where ISBN<15 and IdManga<55 and IdUsuario in (select IdUsuario from compra where ISBN='12')

```

100 %

Resultados Mensajes

	ISBN	IdManga	IdUsuario
1	12	20	5

```

select IdDireccion, IdUsuario
from Direccion
where IdDireccion<15 and IdUsuario in (select IdUsuario from Direccion where IdDireccion='12')

```

100 %

Resultados Mensajes

	IdDireccion	IdUsuario
1	12	5

# CONSULTAS CON OPERACIONES DE CONJUNTOS

--Union, interseccion y complemento--

Tablas que se generan para los ejercicios

The image shows two screenshots of a SQL query execution environment. The first screenshot displays a query that creates a table named ComicsA from the Comics table, filtering for ISBNs between 8 and 9. The second screenshot displays a query that creates a table named ComicsB from the Comics table, filtering for ISBNs between 10 and 12. Both screenshots show the resulting tables with their respective data.

**Query 1: ComicsA**

```
select ISBN, NoComic
into ComicsA
from Comics
where ISBN between 8 and 9;
select * from ComicsA
```

	ISBN	NoComic
1	8	25
2	9	20

**Query 2: ComicsB**

```
select ISBN, NoComic
into ComicsB
from Comics
where ISBN between 10 and 12;
select * from ComicsB
```

	ISBN	NoComic
1	10	22
2	11	21
3	12	23

## UNION

```
select *  
from ComicsA  
union  
select *  
from ComicsB
```

100 % <

Resultados Mensajes

	ISBN	NoComic
1	10	22
2	11	21
3	12	23
4	8	25
5	9	20

```
select NoComic  
from ComicsA  
union  
select NoComic  
from ComicsB
```

100 % <

Resultados Mensajes

	NoComic
1	20
2	21
3	22
4	23
5	25

## INTERSECCION

```
select ISBN
from ComicsA
intersect
select ISBN
from compra
```

100 % <

Resultados Mensajes

	ISBN
1	8
2	9

```
select *
from ComicsA
intersect
select *
from ComicsA
```

100 % <

Resultados Mensajes

	ISBN	NoComic
1	8	25
2	9	20

## COMPLEMENTO

```
select *
from ComicsA
except
select *
from ComicsB
```

100 % <

Resultados Mensajes

	ISBN	NoComic
1	8	25
2	9	20

# CONSULTAS CON PRODUCTO CARTESIANO

--Producto Cartesiano--

100 % <

```
select Contraseña, IdManga, ISBN
from compra cross join usuario where IdManga='20'
```

Resultados Mensajes

	Contraseña	IdManga	ISBN
1	edgr463356	20	10
2	ghfts100	20	10
3	et423	20	10
4	mngs45	20	10
5	asdf24500	20	10
6	edgr463356	20	12
7	ghfts100	20	12
8	et423	20	12
9	mngs45	20	12
10	asdf24500	20	12

100 % <

```
select Nombre, IdManga, ISBN
from compra cross join usuario where IdManga='20'
```

Resultados Mensajes

	Nombre	IdManga	ISBN
1	Eder	20	10
2	Heder	20	10
3	Javier	20	10
4	Luis	20	10
5	Jose	20	10
6	Eder	20	12
7	Heder	20	12
8	Javier	20	12
9	Luis	20	12
10	Jose	20	12

```

select Nombre, Estado, Ciudad
from Direccion cross join usuario where Estado='Puebla'

```

100 %

Resultados Mensajes

	Nombre	Estado	Ciudad
1	Eder	Puebla	Cholula
2	Heder	Puebla	Cholula
3	Javier	Puebla	Cholula
4	Luis	Puebla	Cholula
5	Jose	Puebla	Cholula
6	Eder	Puebla	Cholula
7	Heder	Puebla	Cholula
8	Javier	Puebla	Cholula
9	Luis	Puebla	Cholula
10	Jose	Puebla	Cholula

```

select Nombre, Titulo, Autor
from Manga cross join usuario where Titulo='Naruto'

```

100 %

Resultados Mensajes

	Nombre	Titulo	Autor
1	Eder	Naruto	Takeshi Obata
2	Heder	Naruto	Takeshi Obata
3	Javier	Naruto	Takeshi Obata
4	Luis	Naruto	Takeshi Obata
5	Jose	Naruto	Takeshi Obata
6	Eder	Naruto	Takeshi Obata
7	Heder	Naruto	Takeshi Obata
8	Javier	Naruto	Takeshi Obata
9	Luis	Naruto	Takeshi Obata
10	Jose	Naruto	Takeshi Obata
11	Eder	Naruto	Takeshi Obata
12	Heder	Naruto	Takeshi Obata
13	Javier	Naruto	Takeshi Obata
14	Luis	Naruto	Takeshi Obata
15	Jose	Naruto	Takeshi Obata

```

select IdUsuario, Titulo, NoComic
from Comics cross join compra where NoComic='25'

```

100 %

Resultados Mensajes

	IdUsuario	Titulo	NoComic
1	1	Civil War	25
2	1	Civil War	25
3	2	Civil War	25
4	2	Civil War	25
5	2	Civil War	25
6	3	Civil War	25
7	3	Civil War	25
8	3	Civil War	25
9	4	Civil War	25
10	4	Civil War	25
11	4	Civil War	25
12	5	Civil War	25
13	5	Civil War	25
14	5	Civil War	25



# DEFINICIONES

## SENTENCIAS 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.

## ¿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.

## EXPLICACIÓN DE CONSULTA: LOCAL Y REMOTA

La diferencia existente entre estos dos tipos de consultas, es el hecho de que las consultas locales se ejecutan desde el mismo equipo en que se encuentra la base de datos a utilizar, ya que solo basta en seleccionar las tablas deseadas a modificar o consultar, y escribir las sentencias adecuadas para dichas acciones, mientras que para poder hacer consultas remotas, las cuales se efectúan desde un equipo ajeno al que se localiza la base de datos, se deben de asignar permisos, para poder acceder, ya que el motor de MYSQL es en principio de uso restringido, para seguridad de la información.

## CONCLUSIÓN

Podemos concluir que en 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
- Herramientas tecnológicas: SQL Server.