

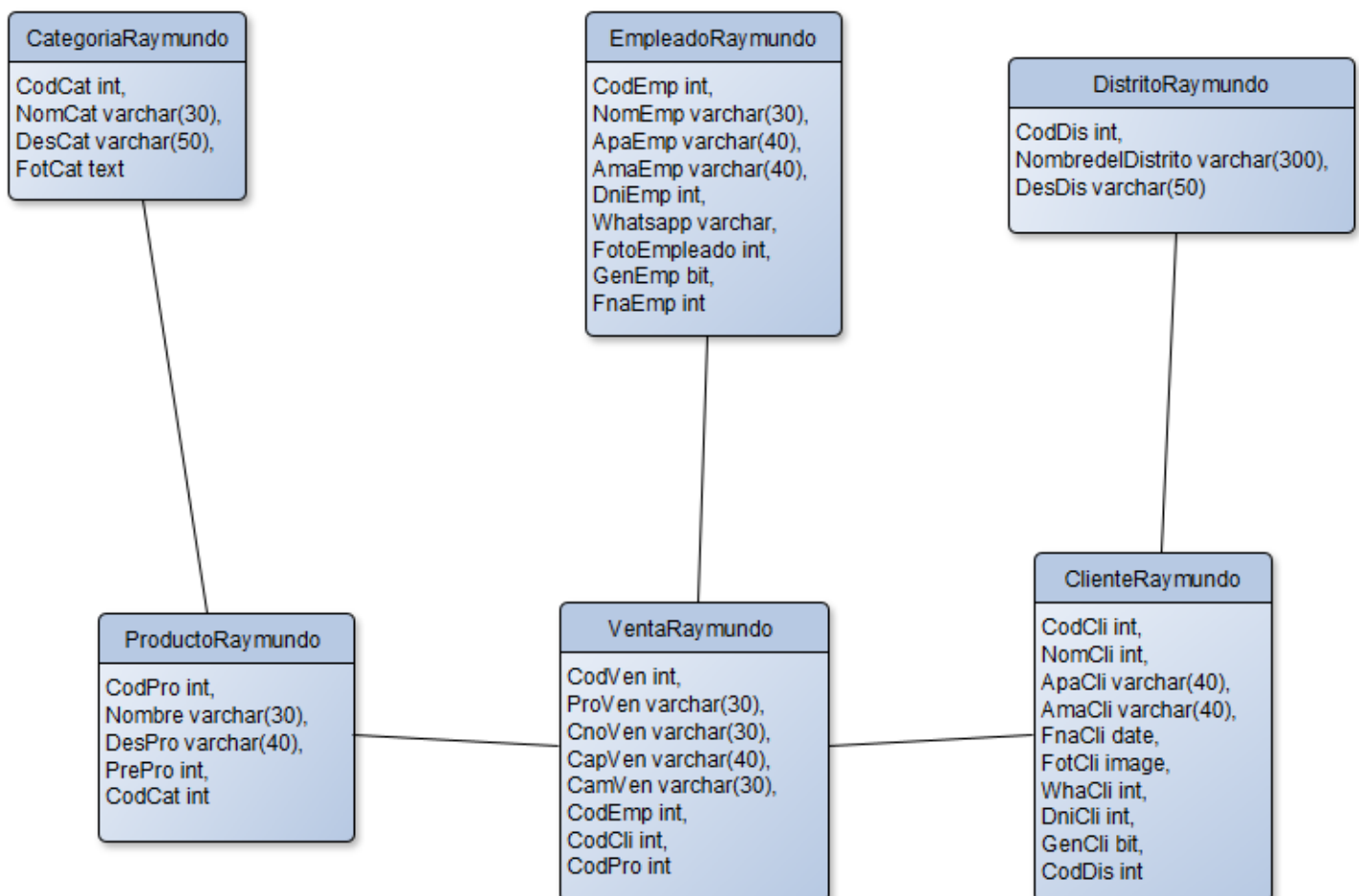
“TEST DE TALENTOS FINAL – RESUELTO”



- **ALUMNA:**
RAYMUNDO ROMERO, MARICARMEN CATALINA
- **CURSO:**
SQL SERVER IMPLEMENTACION
- **PROFESOR:**
FREDY FLORES VENTURA

ANALISIS

- Primero lo que ise es ver el diagrama y analizar de que se trata el base de datos.
- El diagrama era de una venta o tienda.
- Las tablas son:
 - Categoría
 - Producto
 - Empleado
 - Venta
 - Distrito
 - Cliente
- Utilice un programa para la creación del diagrama (yed) para así poder guiarme. Cada tabla les agregue mi apellido al costado como me pidió. También agregue algunos errores en escribir para poder utilizar la sentencia DML.



LA CREACION DE LAS TABLAS

- Creamos nuestra base de datos con el nombre “BDFINALRAYMUNDO”.
- Recordemos que al momento de hacer las tablas y que tenga una relación. Primero tenemos que crear las tablas exteriores. Las tablas exteriores son **CategoriaRaymundo**, **EmpleadoRaymundo** y **DistritoRaymundo**.

```
create table CategoriaRaymundo
(--inicio de la tabla
CodCat int,
NomCat varchar(30),
DesCat varchar(50),
FotCat text--FotCat image
)--final de la tabla
```

```
create table EmpleadoRaymundo
(--inicio de la tabla
CodEmp int,
NomEmp varchar(30),
ApaEmp varchar(40),
AmaEmp varchar(40),
DniEmp int,
Whatsapp varchar,--WhaEmp int
FotoEmpleado int,--FotEmp image
GenEmp bit,
FnaEmp int
)--final de la tabla
```

```
create table DistritoRaymundo
(--inicio de la tabla
CodDis int,
NombredelDistrito varchar(300),--NomDis varchar(30)
DesDis varchar(50)
)--final de la tabla
```

- Después de crear las tablas exteriores, continuamos con las demás tablas que son **ClienteRaymundo** y **ProductoRaymundo**. Pero no la tabla de **VentaRaymundo** porque al momento de crearlo lo vamos a relacionar con dos tablas inmediatamente que son **ClienteRaymundo** y **ProductoRaymundo**. Eso lo dejamos al final.

```

create table ClienteRaymundo
(--inicio de la tabla
CodCli int,
NomCli int,--varchar(30)
ApaCli varchar(40),
AmaCli varchar(40),
FnaCli date,
FotCli image,
WhaCli int,
DniCli int,
GenCli bit,
CodDis int---
)--final de la tabla

create table ProductoRaymundo
(--inicio de la tabla
CodPro int,
Nombre varchar(30),--NomPro varchar(30)--LIST00
DesPro varchar(40),
PrePro int,
CodCat int---
)--final de la tabla

```

- Ejecutamos las dos tablas.
- Ahora vamos a utilizar la sentencia DML porque en nuestra tabla tenemos muchos errores que corregir.
- Primero corregimos la tabla **CategoriaRaymundo**.
- Vemos que el atributo **CodCat** no tiene un **primary key**, así que lo vamos agregar, pero primero lo vamos a eliminar y luego agregar.

```

-----
-----CORREGIR LA TABLA "CategoriaRaymundo"-----
-----
--ELIMINAR "CodCat int"
alter table CategoriaRaymundo--Que Objeto(Tabla)
drop column CodCat--Que Accion (Eliminar)
--AGREGAR " CodCat int primary key identity (20280001,1)"
alter table CategoriaRaymundo--Que Objeto(Tabla)
add CodCat int primary key identity (20280001,1)
-----

```

- La consulta **"alter table CategoriaRaymundo add CodCat int primary key identity (20280001,1)"** realiza la acción de agregar una columna llamada **"CodCat"** a la tabla **"CategoriaRaymundo"**. El tipo de dato de la columna es **"int"** y se define como una clave primaria utilizando la propiedad **"primary key"**. Además, se utiliza la propiedad **"identity"** para generar valores automáticamente para esta columna. El primer valor generado será 20280001 y se incrementará en 1 para cada nueva fila insertada. Esto asegurará que cada valor en la columna **"CodCat"** sea único y se utilice como clave primaria.

- El atributo **FotCat** tiene la identidad incorrecta, debería ser **image**. Así que vamos a corregirlo. Primero eliminamos y después agregamos.

```
--ELIMINAR "FotCat text"
alter table CategoriaRaymundo--Que Objeto(Tabla)
drop column FotCat--Que Accion (Eliminar)
--AGREGAR "FotEmp image default'C:\Users\Administrator\Desktop\img\SinFoto.jpg"
alter table CategoriaRaymundo--Que Objeto(Tabla)
add FotCat image default'C:\Users\Administrator\Desktop\img\SinFoto.jpg'--Que Accion(Adicionar)
-----
```

- La consulta **"alter table CategoriaRaymundo add FotCat image default 'C:\Users\Administrator\Desktop\img\SinFoto.jpg'"** realiza la acción de agregar una columna llamada **"FotCat"** a la tabla **"CategoriaRaymundo"**. El tipo de dato de la columna es **"image"** y se establece un valor predeterminado utilizando la ruta de la imagen **'C:\Users\Administrator\Desktop\img\SinFoto.jpg'**. Esto significa que, si no se proporciona un valor al insertar datos en la tabla, se utilizará automáticamente la imagen especificada como valor predeterminado.
- Ya tenemos corregido de la tabla **CategoriaRaymundo**, así que vamos a verificarlo si tiene las restricciones que añadimos y los tipos.

```
--VERIFICAMOS
sp_help CategoriaRaymundo
```

108 %

Results

Messages

	Name	Owner	Type	Created_datetime
1	CategoriaRaymundo	dbo	user table	2023-07-17 10:48:23.340

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	Trim TrailingBlanks	FixedLenNullInSource	Collation
1	NomCat	varchar	no	30			yes	no	yes	SQL_Latin1_General_CP1_CI_AI
2	DesCat	varchar	no	50			yes	no	yes	SQL_Latin1_General_CP1_CI_AI
3	CodCat	int	no	4	10	0	no	(n/a)	(n/a)	NULL
4	FotCat	image	no	16			yes	(n/a)	(n/a)	NULL

	Identity	Seed	Increment	Not For Replication
1	CodCat	20280001	1	0

	RowGuidCol
1	No rowguidcol column defined.

	Data_located_on_filegroup
1	PRIMARY

	index_name	index_description	index_keys
1	PK_Categori_98E6A20F2A22F1BB	clustered, unique, primary key located on PRIMARY	CodCat

	constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
1	DEFAULT on column FotCat	DF_Categoria__FotCa__3B75D760	(n/a)	(n/a)	(n/a)	(n/a)	(C:\Users\Administrator\Desktop\img\SinFoto.jpg)
2	PRIMARY KEY (clustered)	PK_Categori_98E6A20F2A22F1BB	(n/a)	(n/a)	(n/a)	(n/a)	CodCat

- Ahora corregiremos la tabla **EmpleadoRaymundo** porque tiene varios errores y también falta añadir algunas restricciones.
- En el atributo "CodEmp int" falta añadir primary key, lo vamos a eliminar y lo agregamos con la entidad correcta.

```

-----CORREGIR LA TABLA "EmpleadoRaymundo"-----
--ELIMINAR "CodEmp int"
alter table EmpleadoRaymundo--Que Objeto(Tabla)
drop column CodEmp--Que Accion (Eliminar)
--AGREGAR " CodEmp int primary key identity (20260001,1)"
alter table EmpleadoRaymundo--Que Objeto(Tabla)
add CodEmp int primary key identity (20260001,1)

```

- La consulta "**alter table EmpleadoRaymundo add CodEmp int primary key identity (20260001,1)**" realiza la acción de agregar una columna llamada "**CodEmp**" a la tabla "**EmpleadoRaymundo**". El tipo de dato de la columna es "**int**" y se define como una clave primaria utilizando la propiedad "**primary key**". Además, se utiliza la propiedad "**identity**" para generar valores automáticamente para esta columna. El primer valor generado será 20260001 y se incrementará en 1 para cada nueva fila insertada. Esto asegurará que cada valor en la columna "**CodEmp**" sea único y se utilice como clave primaria.
- En el atributo **DniEmp**, voy a agregarle una restricción.

```

-----AGREGAR "DniEmp" UNA RESTRICCION
alter table EmpleadoRaymundo--Que Objeto(Tabla)
add check(DniEmp>=900000000 and DniEmp<=999999999)
-----

```

- La consulta "**alter table EmpleadoRaymundo add check(DniEmp>=900000000 and DniEmp<=999999999)**" realiza la acción de agregar una restricción de verificación (**check constraint**) a la tabla "**EmpleadoRaymundo**". Esta restricción asegura que los valores en la columna "DniEmp" cumplan con la condición especificada, donde el DNI se encuentre en el rango válido de 900 000 000 a 999 999 999. Si se intenta insertar un valor que no cumpla con esta condición, se mostrará un mensaje de error indicando que el valor del DNI no es válido.

- En el atributo **Whatsapp**, esta mal escrita porque debería estar las tres primeras letras de la tabla y del atributo específico. Así que vamos a eliminar y agregar. También añadiremos una restricción.

```
-----
--ELIMINAR " Whatsapp varchar"
  alter table EmpleadoRaymundo--Que Objeto(Tabla)
    drop column Whatsapp--Que Accion (Eliminar)
--AGREGAR " WhaEmp int"
  alter table EmpleadoRaymundo--Que Objeto(Tabla)
    add WhaEmp int--Que Accion(Adicionar)
--AGREGAR " WhaEmp " UNA RESTRICCION
  alter table EmpleadoRaymundo--Que Objeto(Tabla)
    add check(WhaEmp>=900000000 and WhaEmp<=999999999)
-----
```

- La consulta "**alter table EmpleadoRaymundo add WhaEmp int**" realiza la acción de agregar una nueva columna llamada "**WhaEmp**".
- La consulta "**alter table EmpleadoRaymundo add check (WhaEmp>=900000000 and WhaEmp<=999999999)**" realiza la acción de agregar una restricción de verificación (check constraint) a la columna "**WhaEmp**" en la tabla "**EmpleadoRaymundo**". Esta restricción asegura que los valores en la columna "**WhaEmp**" cumplan con la condición especificada, que el valor de "**WhaEmp**" esté en el rango válido de 900 000 000 a 999 999 999.
- El atributo **FotoEmpleado**, está mal escrita porque debería estar las tres primeras letras de la tabla y del atributo específico. El tipo de dato debería ser **image**. También le voy a agregar una restricción.

```
-----
--ELIMINAR "FotoEmpleado int"
  alter table EmpleadoRaymundo--Que Objeto(Tabla)
    drop column FotoEmpleado--Que Accion (Eliminar)
--AGREGAR "FotEmp image default'C:\Users\Administrator\Desktop\img\SinFoto.jpg'"
  alter table EmpleadoRaymundo--Que Objeto(Tabla)
    add FotEmp image default'C:\Users\Administrator\Desktop\img\SinFoto.jpg'--Que Accion(Adicionar)
-----
```

- La consulta "**alter table EmpleadoRaymundo add FotEmp image default 'C:\Users\Administrator\Desktop\img\SinFoto.jpg'**" realiza la acción de agregar una columna llamada "**FotEmp**" a la tabla "EmpleadoRaymundo". El tipo de dato de la columna es "**image**" y se establece un valor predeterminado utilizando la ruta de la imagen '**C:\Users\Administrator\Desktop\img\SinFoto.jpg**'. Esto significa que, si no se proporciona un valor al insertar datos en la tabla, se utilizará automáticamente la imagen especificada como valor predeterminado.

- El atributo **FnaEmp**, tiene el tipo de dato incorrecto porque el tipo de fecha debe ser **date**. también le vamos agregar una restricción.

```

-----
--ELIMINAR "FnaEmp int"
alter table EmpleadoRaymundo--Que Objeto(Tabla)
drop column FnaEmp--Que Accion (Eliminar)
--AGREGAR " FnaEmp date"
alter table EmpleadoRaymundo--Que Objeto(Tabla)
add FnaEmp date
-----
--AGREGAMOS EN "FnaEmp" UNA RESTRICCION
alter Table EmpleadoRaymundo--Que Objeto(Tabla)
add check(FnaEmp<=getdate())--Que acción (Adicionar)

```

- La consulta **"alter table EmpleadoRaymundo add FnaEmp date"** realiza la acción de agregar una nueva columna llamada **"FnaEmp"** a la tabla **"EmpleadoRaymundo"**.
- La consulta **"alter Table EmpleadoRaymundo add check(FnaEmp<=getdate())"** realiza la acción de agregar una restricción de verificación (check constraint) a la columna **"FnaEmp"** en la tabla **"EmpleadoRaymundo"**. Esta restricción asegura que los valores en la columna **"FnaEmp"** cumplan con la condición especificada, que la fecha de nacimiento (FnaEmp) sea menor o igual a la fecha actual (**getdate()**).
- Ya tenemos corregido de la tabla **EmpleadoRaymundo**, así que vamos verificarlo si tiene las restricciones que añadimos y los tipos.

```
--VERIFICAMOS
sp_help EmpleadoRaymundo
```

Name	Owner	Type	Created_datetime
EmpleadoRaymundo	dbo	user table	2023-07-17 10:48:23.343

Column_name	Type	Computed	Length	Prec	Scale	Nullable	Trim	TrailingBlanks	FixedLen	NullInSource	Collation
1 NomEmp	varchar	no	30			yes	no		yes		SQL_Latin1_General_CP1_CI_AI
2 ApaEmp	varchar	no	40			yes	no		yes		SQL_Latin1_General_CP1_CI_AI
3 AmaEmp	varchar	no	40			yes	no		yes		SQL_Latin1_General_CP1_CI_AI
4 DniEmp	int	no	4	10	0	yes	(n/a)		(n/a)		NULL
5 GenEmp	bit	no	1			yes	(n/a)		(n/a)		NULL
6 CodEmp	int	no	4	10	0	no	(n/a)		(n/a)		NULL
7 WhaEmp	int	no	4	10	0	yes	(n/a)		(n/a)		NULL
8 FotEmp	image	no	16			yes	(n/a)		(n/a)		NULL

Identity	Seed	Increment	Not For Replication
1 CodEmp	20260001	1	0

RowGuidCol
1 No rowguidcol column defined.

Data_located_on_filegroup
1 PRIMARY

index_name	index_description	index_keys
1 PK__Empleado__996E865EFC9C547	clustered, unique, primary key located on PRIMARY	CodEmp

constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
1 CHECK on column DniEmp	CK__EmpleadoR__DniEm__3D5E1FD2	(n/a)	(n/a)	Enabled	Is_For_Replication	((DniEmp)>=(900000000) AND [DniEmp]<=(999999999))
2 CHECK on column FnaEmp	CK__EmpleadoR__FnaEm__403A8C7D	(n/a)	(n/a)	Enabled	Is_For_Replication	((FnaEmp)<=getdate())
3 CHECK on column WhaEmp	CK__EmpleadoR__WhaEm__3E5244...	(n/a)	(n/a)	Enabled	Is_For_Replication	((WhaEmp)>=(900000000) AND [WhaEmp]<=(999999...
4 DEFAULT on column Fot...	DF__EmpleadoR__FotEm__3F466844	(n/a)	(n/a)	(n/a)	(n/a)	('C:\Users\Administrator\Desktop\img\SinFoto.jpg')
5 PRIMARY KEY (clustered)	PK__Empleado__996E865EFC9C547	(n/a)	(n/a)	(n/a)	(n/a)	CodEmp

- Ahora corregiremos la tabla **DistritoRaymundo** porque tiene varios errores y también falta añadir algunas restricciones.

```
-----
-----CORREGIR LA TABLA ""DistritoRaymundo""-----
-----
--ELIMINAR "CodDis int"
alter table DistritoRaymundo--Que Objeto(Tabla)
drop column CodDis--Que Accion (Eliminar)
--AGREGAR " CodDis int primary key identity"
alter table DistritoRaymundo--Que Objeto(Tabla)
add CodDis int primary key identity (20240001,1)
-----
```

- La consulta "**alter table DistritoRaymundo add CodDis int primary key identity (20240001,1)**" realiza la acción de agregar una columna llamada "**CodDis**" a la tabla "**DistritoRaymundo**". El tipo de dato de la columna es "**int**" y se define como una clave primaria utilizando la propiedad "**primary key**". Además, se utiliza la propiedad "**identity**" para generar valores automáticamente para esta columna. El primer valor generado será 20240001 y se incrementará en 1 para cada nueva fila insertada. Esto asegurará que cada valor en la columna "**CodDis**" sea único y se utilice como clave primaria.
- Ahora vamos relacionar la tabla **CategoriaRaymundo** porque tiene una relación uno a muchos con la entidad **ProductoRaymundo**. Un producto pertenece a una categoría, y una categoría puede tener muchos productos.
- Así que vamos a utilizar este comando.

```
--RELACION---
```

```
alter table ProductoRaymundo--Que Objeto(Tabla)ORIGEN
add foreign key(CodCat) references CategoriaRaymundo--Que Accion(Relacionar y/o Referenciar)DESTINO
```

- Ahora vamos a relacionar la tabla **DistritoRaymundo** porque tiene una relación uno a muchos con la entidad **ClienteRaymundo**. Un distrito puede tener varios clientes, pero un cliente reside en un solo distrito.
- Así que vamos a utilizar este comando.

```
--RELACION---
```

```
alter table ClienteRaymundo--Que Objeto(Tabla)ORIGEN
add foreign key(CodDis) references DistritoRaymundo--Que Accion(Relacionar y/o Referenciar)DESTINO
```

- Por último, creamos la tabla **Venta** para que podamos relacionar inmediatamente con las dos tablas que son **ClienteRaymundo** y **ProductoRaymundo**. También agregue una restricción al CodVen.

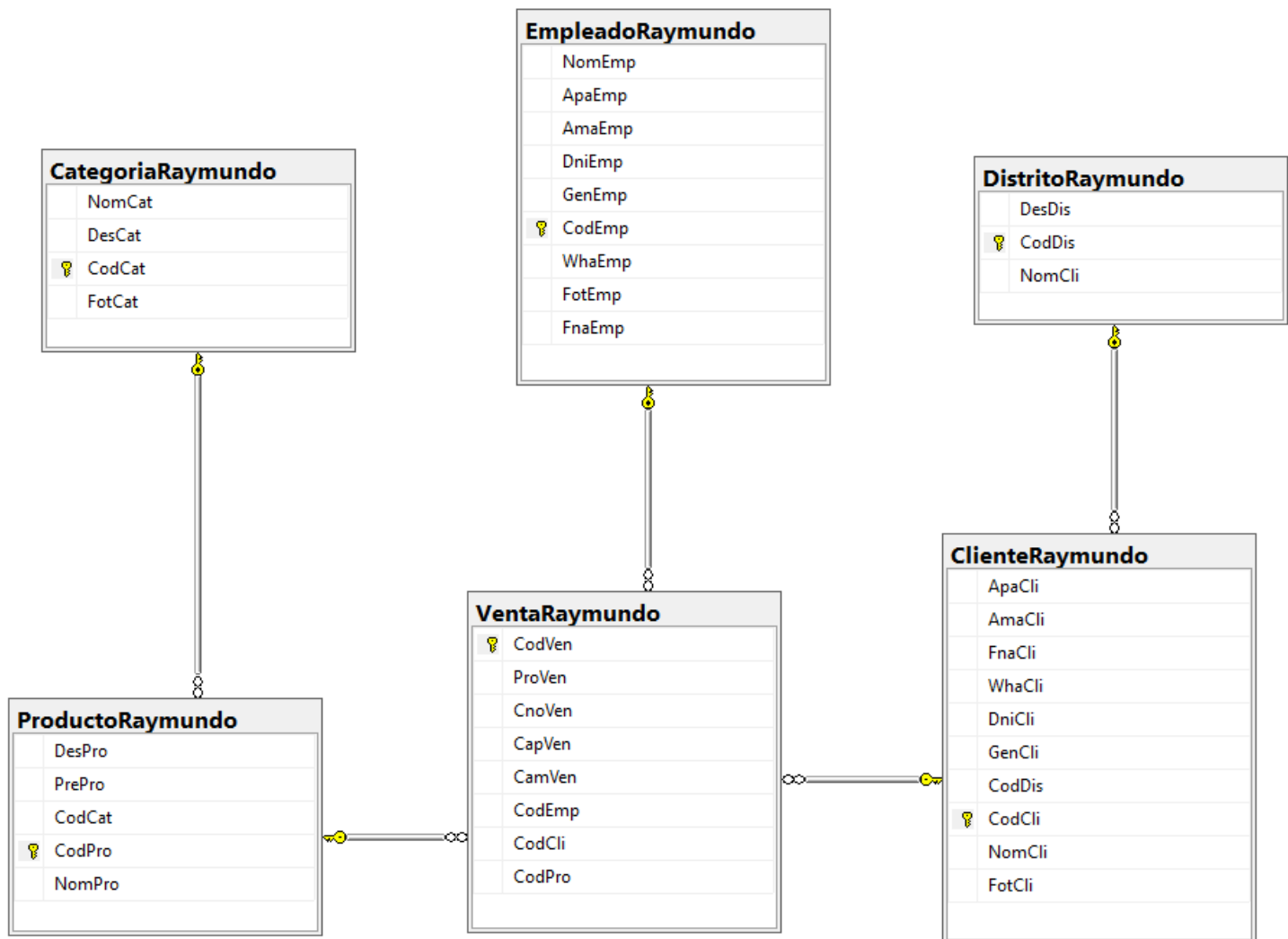
```
create table VentaRaymundo
(--inicio de la tabla
CodVen int primary key identity (20290001,1),
ProVen varchar(30),
CnoVen varchar(30),
CapVen varchar(40),
CamVen varchar(30),
CodEmp int,
CodCli int references ClienteRaymundo,
CodPro int references ProductoRaymundo
)--final de la tabla
```

- Finalmente vamos relacionar la tabla **EmpleadoRaymundo** porque tiene una relación uno a muchos con la entidad **VentaRaymundo**. Un empleado puede realizar varias ventas, pero una venta está asociada a un solo empleado.
- Así que vamos a utilizar este comando.

```
--RELACION--
```

```
alter table VentaRaymundo--Que Objeto(Tabla)ORIGEN
add foreign key(CodEmp) references EmpleadoRaymundo--Que Accion(Relacionar y/o Referenciar)DESTINO
```

ASI QUEDA NUESTRO DIAGRAMA BDFINALRAYMUNDO!



AHORA INSERTAREMOS DATOS A CADA TABLA

- La tabla **CategoriaRaymundo** le vamos agregar 9 datos.

```
insert CategoriaRaymundo(NomCat,DesCat)
values('Embutidos/Quesos', 'Jamonadas, Pates y Quesos'),('Juguetes', 'Muñecas, Coches, Lego y Carros'),('Libros', 'Romance, Terror, Comedia, Misterio y Cocina'),('Ropa', 'Polo, Pantalon, Falda, Vestido y Camisa'),('Bebidas', 'Gaseosa, Agua, Alcohol, Energizantes y Cervezas')

insert CategoriaRaymundo(NomCat,DesCat)
values('Electrodoméstico', 'Televisores, Refrigeradoras y Equipos de Sonido'),('Frutas', 'Frutas de Estacion')

insert CategoriaRaymundo(NomCat,DesCat)
values('Abarrotes', 'Viveres y/o alimentos envasados'),('Informatica', 'Impresoras, Laptops, Suministros y Partes')
```

- Aquí utilice este comando para poder ver la tabla **CategoriaRaymundo** y sus datos.

select*
from CategoriaRaymundo

98 %

Results Messages

	NomCat	DesCat	CodCat	FotCat
1	Embutidos/Quesos	Jamonadas, Pates y Quesos	20280012	0x433A5C55736572735C41646D696E6973747261746F725C...
2	Juguetes	Muñecas, Coches, Lego y Carros	20280013	0x433A5C55736572735C41646D696E6973747261746F725C...
3	Libros	Romance, Terror, Comedia, Misterio y Cocina	20280014	0x433A5C55736572735C41646D696E6973747261746F725C...
4	Ropa	Polo, Pantalon, Falda, Vestido y Camisa	20280015	0x433A5C55736572735C41646D696E6973747261746F725C...
5	Bebidas	Gaseosa, Agua, Alcohol, Energizantes y Cervezas	20280016	0x433A5C55736572735C41646D696E6973747261746F725C...
6	Electrodoméstico	Televisores, Refrigeradoras y Equipos de Sonido	20280024	0x433A5C55736572735C41646D696E6973747261746F725C...
7	Frutas	Frutas de Estacion	20280025	0x433A5C55736572735C41646D696E6973747261746F725C...
8	Abarrotes	Viveres y/o alimentos envasados	20280028	0x433A5C55736572735C41646D696E6973747261746F725C...
9	Informatica	Impresoras, Laptops, Suministros y Partes	20280029	0x433A5C55736572735C41646D696E6973747261746F725C...

- La tabla **EmpleadoRaymundo** le vamos agregar 9 datos.

--TABLA EMPLEADORAYMUNDO---

```
insert EmpleadoRaymundo (NomEmp, ApaEmp, AmaEmp, DniEmp, WhaEmp, GenEmp, FnaEmp)
values('Juan', 'García', 'Pérez', 912345678, 912345678, 0, '1990-01-15'),('María', 'López', 'Gómez',
923456789, 923456789, 1, '1992-05-20'),('Pedro', 'Martínez', 'Rodríguez', 934567890, 934567890, 0,
'1995-08-10')

insert EmpleadoRaymundo (NomEmp, ApaEmp, AmaEmp, DniEmp, WhaEmp, GenEmp, FnaEmp)
values('Laura', 'Hernández', 'Sánchez', 945678901, 945678901, 1, '1998-02-28'),('Carlos',
'González', 'Lara', 956789012, 956789012, 0, '1994-11-12'),('Ana', 'Torres', 'Vargas', 967890123,
967890123, 1, '1991-07-05'),('Luis', 'Ramírez', 'Morales', 978901234, 978901234, 0, '1997-03-25')

insert EmpleadoRaymundo (NomEmp, ApaEmp, AmaEmp, DniEmp, WhaEmp, GenEmp, FnaEmp)
values('Sofía', 'Soto', 'Flores', 989012345, 989012345, 1, '1993-09-17'),('Diego', 'Ortega',
'Guerrero', 990123456, 990123456, 0, '1996-06-08')
```

- Aquí utilice este comando para poder ver la tabla **EmpleadoRaymundo** y sus datos.

	NomEmp	ApaEmp	AmaEmp	DniEmp	GenEmp	CodEmp	WhaEmp	FotEmp	FnaEmp
1	Juan	García	Pérez	912345678	0	20260004	912345678	0x433A5C55736572735C41646D696E6973747261746F725C...	1990-01-15
2	María	López	Gómez	923456789	1	20260005	923456789	0x433A5C55736572735C41646D696E6973747261746F725C...	1992-05-20
3	Pedro	Martínez	Rodríguez	934567890	0	20260006	934567890	0x433A5C55736572735C41646D696E6973747261746F725C...	1995-08-10
4	Laura	Hernández	Sánchez	945678901	1	20260007	945678901	0x433A5C55736572735C41646D696E6973747261746F725C...	1998-02-28
5	Carlos	González	Lara	956789012	0	20260008	956789012	0x433A5C55736572735C41646D696E6973747261746F725C...	1994-11-12
6	Ana	Torres	Vargas	967890123	1	20260009	967890123	0x433A5C55736572735C41646D696E6973747261746F725C...	1991-07-05
7	Luis	Ramírez	Morales	978901234	0	20260010	978901234	0x433A5C55736572735C41646D696E6973747261746F725C...	1997-03-25
8	Sofía	Soto	Flores	989012345	1	20260011	989012345	0x433A5C55736572735C41646D696E6973747261746F725C...	1993-09-17
9	Diego	Ortega	Guerrero	990123456	0	20260012	990123456	0x433A5C55736572735C41646D696E6973747261746F725C...	1996-06-08

- La tabla **DistritoRaymundo** le vamos agregar 9 datos.

```
--TABLA DISTRITORAYMUNDO
```

```
insert DistritoRaymundo (NomDis, DesDis)
values('Miraflores', 'Distrito turístico')
```

```
insert DistritoRaymundo (NomDis, DesDis)
values('San Isidro', 'Distrito financiero'),('Barranco', 'Distrito bohemio y artístico ubicado en Lima')
```

```
insert DistritoRaymundo (NomDis, DesDis)
values('San Borja', 'Distrito residencial'),('Surco', 'Distrito comercial ubicado en Lima.'),('Callao', 'Ubicado en la costa central de Perú.')
```

```
insert DistritoRaymundo (NomDis, DesDis)
values('Chorrillos', 'Distrito costero ubicado en Lima.'),('La Molina', 'Distrito residencial y universitario.'),('Lince', 'Distrito céntrico y residencial.')
```

- Aquí utilice este comando para poder ver la tabla **DistritoRaymundo** y sus datos.

```
--Mostrar
select *
from DistritoRaymundo
```

	DesDis	CodDis	NomDis
1	Distrito turístico	20240003	Miraflores
2	Distrito financiero	20240005	San Isidro
3	Distrito bohemio y artístico ubicado en Lima	20240006	Barranco
4	Distrito residencial	20240010	San Borja
5	Distrito comercial ubicado en Lima.	20240011	Surco
6	Ubicado en la costa central de Perú.	20240012	Callao
7	Distrito costero ubicado en Lima.	20240013	Chorrillos
8	Distrito residencial y universitario.	20240014	La Molina
9	Distrito céntrico y residencial.	20240015	Lince

- La tabla **ClienteRaymundo** le vamos agregar 9 datos.

---TABLA CLIENTERAYMUNDO

```
insert ClienteRaymundo (NomCli, ApaCli, AmaCli, FnaCli, WhaCli, DniCli, GenCli, CodDis)
values('Martha', 'Zuñiga', 'Pérez', '1990-01-15', 912345678, 902349988, 0, 20240003),('Stiven',
'López', 'Entrada', '1992-05-20', 923456789, 922455599, 1, 20240005),('Larry', 'Rios', 'Farias',
'1995-08-10', 934567890, 938867890, 0, 20240011)

insert ClienteRaymundo (NomCli, ApaCli, AmaCli, FnaCli, WhaCli, DniCli, GenCli, CodDis)
values('Helen', 'Chau', 'Romero', '1998-02-28', 945678901, 999678801, 1, 20240012),('Carlos',
'González', 'Lara', '1994-11-12', 956789012, 957777012, 0, 20240014),('Abigail', 'Torres',
'Vargas', '1991-07-05', 967890123, 967890123, 1, 20240015),('Leonardo', 'Caja', 'Morales', '1997-
03-25', 978901234, 975551234, 0, 20240006),('Julieta', 'Leon', 'Acosta', '1993-09-17', 989012345,
900012225, 1, 20240006),('Mathias', 'Baldeon', 'Campos', '1996-06-08', 990123456, 955663226, 0,
20240010)
```

- Aquí utilice este comando para poder ver la tabla **ClienteRaymundo** y sus datos.

	ApaCli	AmaCli	FnaCli	WhaCli	DniCli	GenCli	CodDis	CodCli	NomCli	FotCli
1	Zuñiga	Pérez	1990-01-15	912345678	902349988	0	20240003	20230001	Martha	0x433A5C55736572735C41646D696E6973747261746F725C...
2	López	Entrada	1992-05-20	923456789	922455599	1	20240005	20230002	Stiven	0x433A5C55736572735C41646D696E6973747261746F725C...
3	Rios	Farias	1995-08-10	934567890	938867890	0	20240011	20230003	Lary	0x433A5C55736572735C41646D696E6973747261746F725C...
4	Chau	Romero	1998-02-28	945678901	999678801	1	20240012	20230004	Helen	0x433A5C55736572735C41646D696E6973747261746F725C...
5	González	Lara	1994-11-12	956789012	957777012	0	20240014	20230005	Carlos	0x433A5C55736572735C41646D696E6973747261746F725C...
6	Torres	Vargas	1991-07-05	967890123	967890123	1	20240015	20230006	Abigail	0x433A5C55736572735C41646D696E6973747261746F725C...
7	Caja	Morales	1997-03-25	978901234	975551234	0	20240006	20230007	Leon...	0x433A5C55736572735C41646D696E6973747261746F725C...
8	Leon	Acosta	1993-09-17	989012345	900012225	1	20240006	20230008	Julieta	0x433A5C55736572735C41646D696E6973747261746F725C...
9	Baldeon	Campos	1996-06-08	990123456	955663226	0	20240010	20230009	Math...	0x433A5C55736572735C41646D696E6973747261746F725C...

- La tabla **ProductoRaymundo** le vamos agregar 9 datos.

```
insert ProductoRaymundo (NomPro, DesPro, PrePro, CodCat)
values('Agua San Mateo de 1.5L', 'La más ecológica', 12.99, 20280005),('Lavadora LG 16KG', 'Dura
mas tiempo que los demas', 225.99, 20280006)

insert ProductoRaymundo (NomPro, DesPro, PrePro, CodCat)
values('Polo Adiddas', 'Buena calidad para los demas.', 55.99, 20280004),('Muñeca Sirenita',
'Tiene unas aletas de colores', 29.99, 20280002),('Manzana Roja', 'Tiempo de comer saludable.',
4.99, 20280007),('Mandarina', 'Delicioso y sin pepas.', 3.99, 20280007)

insert ProductoRaymundo (NomPro, DesPro, PrePro, CodCat)
values('Energizante RED BULL 250ml', 'Buena para el deporte.', 8.99, 20280005),('Aceite CAPRI
1L', 'Calidad vegetal.', 7.99, 20280008),('Chirimoya Cumbe', 'Tiempo de comer dulce.', 2.99,
20280007)
```

- Aquí utilice este comando para poder ver la tabla **ProductoRaymundo** y sus datos.

Results		Messages		
	NomCat	DesCat	CodCat	FotCat
1	Embutidos/Quesos	Jamonadas, Pates y Quesos	20280001	0x433A5C55736572735C41646D696E6973747261746F725C...
2	Juguetes	Muñecas, Coches, Lego y Carros	20280002	0x433A5C55736572735C41646D696E6973747261746F725C...
3	Libros	Romance, Terror, Comedia, Misterio y Cocina	20280003	0x433A5C55736572735C41646D696E6973747261746F725C...
4	Ropa	Polo, Pantalón, Falda, Vestido y Camisa	20280004	0x433A5C55736572735C41646D696E6973747261746F725C...
5	Bebidas	Gaseosa, Agua, Alcohol, Energizantes y Cervezas	20280005	0x433A5C55736572735C41646D696E6973747261746F725C...
6	Electrodoméstico	Televisores, Refrigeradoras y Equipos de Sonido	20280006	0x433A5C55736572735C41646D696E6973747261746F725C...
7	Frutas	Frutas de Estacion	20280007	0x433A5C55736572735C41646D696E6973747261746F725C...
8	Abarotes	Viveres y/o alimentos envasados	20280008	0x433A5C55736572735C41646D696E6973747261746F725C...
9	Informatica	Impresoras, Laptops, Suministros y Partes	20280009	0x433A5C55736572735C41646D696E6973747261746F725C...

- La tabla **VentaRaymundo** le vamos agregar 9 datos.

----TABLA VENTARAYMUNDO

```

insert VentaRaymundo (ProVen, CnoVen, CapVen, CamVen, CodEmp, CodCli, CodPro)
values('Mandarina', 'Leonardo', 'Caja', 'Morales', 20260006, 20230004, 20270007), ('Manzana
Roja', 'Martha', 'Zuñiga', 'Pérez', 20260008, 20230001, 20270006), ('Aceite CAPRI 1L',
'Abigail', 'Torres', 'Vargas', 20260009, 20230006, 20270012)

insert VentaRaymundo (ProVen, CnoVen, CapVen, CamVen, CodEmp, CodCli, CodPro)
values('Chirimoya Cumbe', 'Leonardo', 'Caja', 'Morales', 20260006, 20230004,
20270013), ('Mandarina', 'Martha', 'Zuñiga', 'Pérez', 20260006, 20230001, 20270007), ('Lavadora LG
16KG', 'Martha', 'Zuñiga', 'Pérez', 20260009, 20230001, 20270003), ('Agua San Mateo de 1.5L',
'Leonardo', 'Caja', 'Morales', 20260008, 20230004, 20270002), ('Agua San Mateo de 1.5L',
'Abigail', 'Torres', 'Vargas', 20260009, 20230006, 20270002)

insert VentaRaymundo (ProVen, CnoVen, CapVen, CamVen, CodEmp, CodCli, CodPro)
values('Energizante RED BULL 250ml', 'Julietta', 'Leon', 'Acosta', 20260001, 20230008,
20270011), ('Polo Adiddas', 'Julietta', 'Leon', 'Acosta', 20260001, 20230008, 20270004), ('Agua San
Mateo de 1.5L', 'Larry', 'Rios', 'Farias', 20260002, 20230003, 20270002), ('Muñeca Sirenita',
'Larry', 'Rios', 'Farias', 20260002, 20230003, 20270005), ('Muñeca Sirenita', 'Mathias',
'Baldeon', 'Campos', 20260006, 20230009, 20270005), ('Energizante RED BULL 250ml', 'Mathias',
'Baldeon', 'Campos', 20260006, 20230009, 20270011)

insert VentaRaymundo (ProVen, CnoVen, CapVen, CamVen, CodEmp, CodCli, CodPro)
values('Aceite CAPRI 1L', 'Carlos', 'González', 'Lara', 20260006, 20230005, 20270012), ('Agua
San Mateo de 1.5L', 'Carlos', 'González', 'Lara', 20260009, 20230005, 20270002), ('Lavadora LG
16KG', 'Stiven', 'López', 'Entrada', 20260001, 20230002, 20270003), ('Manzana Roja', 'Stiven',
'López', 'Entrada', 20260001, 20230002, 20270006), ('Polo Adiddas', 'Stiven', 'López', 'Entrada',
20260009, 20230002, 20270004), ('Muñeca Sirenita', 'Stiven', 'López', 'Entrada', 20260002,
20230002, 20270005)

```

- Aquí utilice este comando para poder ver la tabla **VentaRaymundo** y sus datos.

```
--Mostrar
```

```
select*
from VentaRaymundo
```

98 %

Results Messages

	CodVen	ProVen	CnoVen	CapVen	CamVen	CodEmp	CodCli	CodPro
1	20290001	Mandarina	Leonardo	Caja	Morales	20260006	20230004	20270007
2	20290002	Manzana Roja	Martha	Zuñiga	Pérez	20260008	20230001	20270006
3	20290003	Aceite CAPRI 1L	Abigail	Torres	Vargas	20260009	20230006	20270012
4	20290004	Chirimoya Cumbe	Leonardo	Caja	Morales	20260006	20230004	20270013
5	20290005	Mandarina	Martha	Zuñiga	Pérez	20260006	20230001	20270007
6	20290006	Lavadora LG 16KG	Martha	Zuñiga	Pérez	20260009	20230001	20270003
7	20290007	Agua San Mateo de 1.5L	Leonardo	Caja	Morales	20260008	20230004	20270002
8	20290008	Agua San Mateo de 1.5L	Abigail	Torres	Vargas	20260009	20230006	20270002
9	20290009	Energizante RED BULL 250ml	Julieta	Leon	Acosta	20260001	20230008	20270011
10	20290010	Polo Adiddas	Julieta	Leon	Acosta	20260001	20230008	20270004
11	20290011	Agua San Mateo de 1.5L	Lary	Rios	Farias	20260002	20230003	20270002
12	20290012	Muñeca Sirenita	Lary	Rios	Farias	20260002	20230003	20270005
13	20290013	Muñeca Sirenita	Mathias	Baldeon	Campos	20260006	20230009	20270005
14	20290014	Energizante RED BULL 250ml	Mathias	Baldeon	Campos	20260006	20230009	20270011
15	20290015	Aceite CAPRI 1L	Carlos	González	Lara	20260006	20230005	20270012
16	20290016	Agua San Mateo de 1.5L	Carlos	González	Lara	20260009	20230005	20270002
17	20290017	Lavadora LG 16KG	Stiven	López	Entrada	20260001	20230002	20270003
18	20290018	Manzana Roja	Stiven	López	Entrada	20260001	20230002	20270006
19	20290019	Polo Adiddas	Stiven	López	Entrada	20260009	20230002	20270004
20	20290020	Muñeca Sirenita	Stiven	López	Entrada	20260002	20230002	20270005

CONSULTA

- Lo primero para hacer la consulta es ver mis tablas mediante con el diagrama que tengo. Entonces vi que hay 4 tablas relacionadas a la consulta pedido. Respondí estas preguntas para poder realizar la consulta.

```
-----  
--LISTA DE CONSULTAS  
--1. Una consulta de los nombres de los clientes, codigo del producto, precio del producto y  
el nombre del empleado  
  
--Que tablas son?  
ClienteRaymundo, VentaRaymundo, ProductoRaymundo, EmpleadoRaymundo  
--Están relacionadas?  
si  
--Cuáles son las columnas en común?  
CodPro, CodCli  
--Unir y Habilitar la columna en común?  
  
select ClienteRaymundo.NomCli as Nombre, ProductoRaymundo.PrePro as Precio,  
ProductoRaymundo.CodCat as CodigoProducto, EmpleadoRaymundo.NomEmp as Empleado  
from ClienteRaymundo  
JOIN VentaRaymundo  
on ClienteRaymundo.CodCli = VentaRaymundo.CodCli  
JOIN ProductoRaymundo  
on VentaRaymundo.CodPro = ProductoRaymundo.CodPro  
JOIN EmpleadoRaymundo  
on VentaRaymundo.CodEmp = EmpleadoRaymundo.CodEmp
```

- Resultado del comando realizado.

Results		Messages		
	Nombre	Precio	CodigoProducto	Empleado
1	Helen	3	20280007	Ana
2	Martha	4	20280007	Sofía
3	Abigail	7	20280008	Diego
4	Helen	2	20280007	Ana
5	Martha	3	20280007	Ana
6	Martha	225	20280006	Diego
7	Helen	12	20280005	Sofía
8	Abigail	12	20280005	Diego
9	Julieta	8	20280005	Juan
10	Julieta	55	20280004	Juan
11	Lary	12	20280005	María
12	Lary	29	20280002	María
13	Mathias	29	20280002	Ana
14	Mathias	8	20280005	Ana
15	Carlos	7	20280008	Ana
16	Carlos	12	20280005	Diego
17	Stiven	225	20280006	Juan
18	Stiven	4	20280007	Juan
19	Stiven	55	20280004	Diego
20	Stiven	29	20280002	María

PA

- Este comando ejecutará el procedimiento almacenado y mostrará los registros de **ClienteRaymundo** en la tabla especificada.

```
--Crear PA para listar los Clientes
create proc PAListarClienteNombre
as
begin --inicia
--debemos utilizar cualquier comando SQL
select*
from ClienteRaymundo

end--termina el PA

--Luego de crear para ejecutar
exec PAListarClienteNombre
```

- Esto es el resultado de **"PAListarClienteNombre"**

	Results	Messages								
	ApaCli	AmaCli	FnaCli	WhaCli	DniCli	GenCli	CodDis	CodCli	NomCli	FotCli
1	Zuñiga	Pérez	1990-01-15	912345678	902349988	0	20240003	20230001	Martha	0x433A5C55736572735C41646D696E6973747261746F725C...
2	López	Entrada	1992-05-20	923456789	922455599	1	20240005	20230002	Stiven	0x433A5C55736572735C41646D696E6973747261746F725C...
3	Rios	Farias	1995-08-10	934567890	938867890	0	20240011	20230003	Lamy	0x433A5C55736572735C41646D696E6973747261746F725C...
4	Chau	Romero	1998-02-28	945678901	999678801	1	20240012	20230004	Helen	0x433A5C55736572735C41646D696E6973747261746F725C...
5	González	Lara	1994-11-12	956789012	957777012	0	20240014	20230005	Carlos	0x433A5C55736572735C41646D696E6973747261746F725C...
6	Tomes	Vargas	1991-07-05	967890123	967890123	1	20240015	20230006	Abigail	0x433A5C55736572735C41646D696E6973747261746F725C...
7	Caja	Morales	1997-03-25	978901234	975551234	0	20240006	20230007	Leonardo	0x433A5C55736572735C41646D696E6973747261746F725C...
8	Leon	Acosta	1993-09-17	989012345	900012225	1	20240006	20230008	Julieta	0x433A5C55736572735C41646D696E6973747261746F725C...
9	Baldeon	Campos	1996-06-08	990123456	955663226	0	20240010	20230009	Mathias	0x433A5C55736572735C41646D696E6973747261746F725C...

- El código que muestra, es una alteración del procedimiento almacenado "PAListarClienteNombre". En esta versión modificada, se ha agregado un parámetro llamado "@Nom" de tipo varchar(20). Luego, se utiliza este parámetro en la cláusula **WHERE** de la consulta **SELECT** para filtrar los resultados de la tabla "ClienteRaymundo" según el nombre de cliente ingresado.

```
alter proc PAListarClienteNombre
@Nom varchar(20)--Ingresamos el parametro
as
begin --inicia el PA
--debemos utilizar cualquier comando SQL
select*
from ClienteRaymundo
where NomCli =@Nom
end--termina el PA

--Ejecutamos el PA e ingreando el WhaCli
exec PAListarClienteNombre
```

- Esto ejecutará el procedimiento almacenado "PAListarClienteNombre" con el valor proporcionado para el parámetro "@Nom", y mostrará los registros de clientes que cumplan con el filtro.

```
--Para mostrar el contenido de un PA
sp_helptext PAListarClienteNombre
```

Results		Messages
	Text	
1	CREATE proc PAListarCliente	
2	@Wha varchar(15)--Ingresamos el parametro	
3	as	
4	begin -inicia el PA	
5	-debemos utilizar cualquier comando SQL	
6	select*	
7	from ClienteRaymundo	
8	where WhaCli =@Wha	
9	end-termina el PA	
10		
11	-Ejecutamos el PA e ingreando la region desde ...	
12	exec PAListarCliente	

- Para eliminar un procedimiento almacenado, se utiliza la instrucción **DROP PROCEDURE**. En este caso para eliminar el procedimiento almacenado "**PAListarCliente**", este código es:

```
--COMO ELIMINAR  
drop proc PAListarCliente
```

MUCHAS GRACIAS