

## Actividad | #2 | Consultas

Ingeniería en Desarrollo de Software

---



TUTOR: Miguel Angel Rodríguez Vega

---

ALUMNO: Edgar Enrique Cuamea Ochoa

---

FECHA: 25 de mayo del 2024

---

## Contenido

Introducción. ....	3
Descripción. ....	4
Justificación. ....	5
Desarrollo.....	6
Conclusión. ....	17
Referencias.....	18

### **Introducción.**

En esta actividad veremos las consultas que realizaremos en las bases de datos que importamos en la actividad anterior por lo que utilizaremos a través del lenguaje SQL estaremos realizando diferentes tipos de consultas, tales como cantidad de clientes que se registraron en un año determinado así como las compras realizadas por distintos clientes en cierto periodo de tiempo o consultar cual fue el producto más vendido en un periodo de tiempo de varios meses así como ver los encargados de las diferentes tiendas así como ordenarlos alfabéticamente, o saber cuáles son los empleados de una sucursal en concreto, ver categorías, etc. Esta actividad estará enfocada en las consultas que realizaremos en nuestra base de datos por lo que en el proceso de las consultas agregaremos capturas de las consultas realizadas, así como una descripción de que información queremos consultar, así como el código escrito en la consola utilizando en lenguaje SQL.

### **Descripcion.**

Descargaremos e importaremos una base de datos por lo que analizaremos de qué forma está construida esta base de datos para poder realizar consultas en esta base de datos con diferentes tablas y atributos por lo que realizaremos consultas de datos en algunas tablas para poder así identificar las relaciones de estas tablas ya que diferenciaremos las llaves primarias y llaves foráneas de la base de datos importada, además de realizar consultas de diferentes tablas tales como clientes y productos así como poder ver diferentes tipos de consultas ya que realizaremos consultas entre algunos periodos de tiempo además de ver algunas compras que realizaron los clientes, ver que producto fue el más vendido o cual es el producto del que tiene más stock, realizaremos las consultas utilizando el lenguaje SQL para poder visualizar los datos que queremos consultar, además de revisar la sintaxis del código ya que si nos equivocamos en un símbolo o alguna letra así como el nombre de las tablas, puede hacer que el código no funcione correctamente.

### **Justificación.**

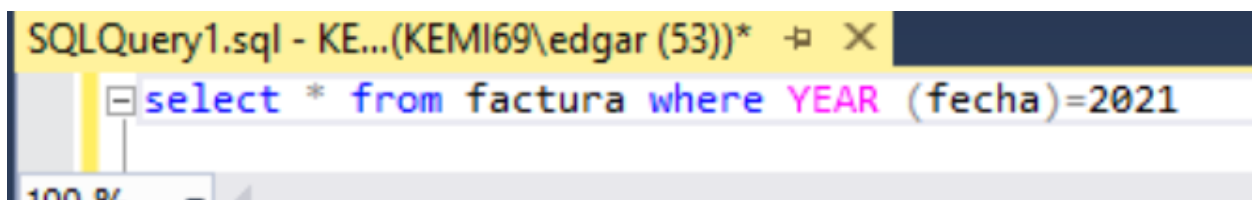
Utilizaremos SQL Server para la creación y administración de las bases de datos ya que este programa requiere el uso del lenguaje SQL para la manipulación de datos en una base de datos local por lo que podremos realizar la actividad en esta plataforma ya que SQL Server nos permite exportar e importar bases de datos ya creadas por lo que utilizando SQL Server, descargaremos una base de datos de una boutique que nos proporciona el pdf de las instrucciones de la actividad por lo que importaremos la base de datos y verificaremos que se importe de manera correcta ya que pueden ser de distintas versiones del programa, lo que lo hace una forma fácil de poder solucionar y fácil de utilizar bases de datos de diferentes versiones, además de poder reconocer las relaciones de las tablas de las bases de datos así como el uso de las llaves foráneas y llaves primarias ya que realizaremos diferentes consultas dentro de las bases de datos así poder aprender a utilizar este programa de administración de las bases de datos.

### Desarrollo.

se desea conocer la siguiente información para la creación de consultas:

- La cantidad de clientes en el año 2021.
- La cantidad de clientes en lo que va del año 2022.
- Los clientes que tuvieron en diciembre de 2021.
- ¿Qué compras han realizado algunos clientes en específico?
- ¿Qué producto es el que más ventas ha tenido?
- ¿Qué producto tiene más cantidad en stock?
- Ordenar por fecha las compras que ha habido en la tienda.
- Ordenar alfabéticamente los nombres de los clientes de la tienda.
- Seleccionar cuántos productos hay en cada categoría: falda, pantalón, chamarra, zapatos y accesorios.
- ¿Cuáles son los encargados en cada sucursal de Akira's Boutique?
- ¿Cuáles son los empleados que trabajan en la sucursal de Akira's Boutique sucursal Constitución?
- ¿Cuáles clientes son mayores de 30 años?

Realizaremos la primera consulta en nuestra base de datos donde verificaremos la cantidad de clientes en el año 2021.



The screenshot shows a window titled "SQLQuery1.sql - KE...(KEMI69\edgar (53))\*". The query text is: `select * from factura where YEAR (fecha)=2021`. The window has a standard toolbar with icons for saving, undo, redo, and other functions. The status bar at the bottom shows "100 %".

Seleccionamos todos los datos de la tabla factura donde condicionamos que seleccionaremos datos de tipo fecha, en este caso el año, por lo que entre paréntesis seleccionamos de que tabla o atributo

estaremos filtrando la información por lo que agregamos un igual después de el paréntesis e indicamos cual es el año que queremos que nos muestre por lo que al ejecutar el comando nos muestra la siguiente tabla donde nos muestra una lista del id de la factura así como el id del cliente y del detalle, por lo que vemos que tenemos un total de 23 clientes en el año 2021

	id_factura	id_cliente	fecha	id_detalle
1	1	9	2021-12-22	1
2	2	9	2021-10-11	2
3	3	4	2021-12-07	3
4	4	4	2021-08-24	4
5	5	1	2021-09-24	5
6	8	0	2021-12-28	8
7	11	2	2021-07-13	11
8	14	6	2021-12-22	14
9	15	5	2021-11-18	15
10	17	7	2021-10-12	17
11	18	7	2021-12-30	18
12	20	8	2021-12-21	20
13	21	26	2021-10-08	21
14	23	31	2021-07-16	23
15	24	35	2021-09-23	24
16	25	25	2021-06-14	25
17	26	29	2021-08-19	26
18	27	27	2021-11-10	27
19	31	12	2021-12-07	31
20	34	17	2021-11-11	34
21	35	31	2021-11-16	35
22	37	23	2021-11-17	37
23	39	26	2021-12-14	39

Seleccionamos los clientes de 2022 (hasta el momento) en base a la fecha de la factura.

Utilizaremos un código parecido al anterior, solo modificando algunos datos por lo que ingresamos el siguiente código

```
select * from factura where (fecha) > '2022-01-01'
```

En esta línea seleccionamos todos los datos de la tabla factura donde la fecha sea mayor al primero de enero de el año 2022 y al ejecutar el comando vemos la siguiente tabla:

	id_factura	id_cliente	fecha	id_detalle
1	6	1	2022-01-08	6
2	9	3	2022-01-11	9
3	10	3	2022-01-16	10
4	12	2	2022-01-05	12
5	13	6	2022-01-12	13
6	16	5	2022-01-09	16
7	19	8	2022-01-17	19
8	22	30	2022-01-19	22
9	28	24	2022-01-22	28
10	29	26	2022-01-11	29
11	30	36	2022-01-04	30
12	32	14	2022-01-06	32
13	33	24	2022-01-03	33
14	36	36	2022-01-10	36
15	38	29	2022-01-03	38
16	40	18	2022-01-20	40

Donde tenemos un total de 16 facturas realizadas en el año 2022 hasta la fecha, cada una de estas facturas contiene el id de la factura así como un id de el cliente que realizó la compra además de el detalle de la compra así como su respectiva fecha de compra.

Seleccionamos los clientes de diciembre de 2021, en este caso utilizaremos el siguiente código:

```

SQLQuery1.sql - KE...(KEMI69\edgar (83))*  X
select * from factura
where (fecha) > '2021-12-01' and (fecha) < '2022-01-01'

```

En este caso estamos seleccionando todos los datos de la tabla factura, donde establecemos los siguientes parámetros, estaremos seleccionando todas las facturas donde la fecha sea mayor al primero de diciembre del año 2021 y menor a la fecha de primero de enero del año 2022 por lo que al ejecutar el código nos muestra los siguientes datos donde podremos ver el total de facturas realizadas en el mes de diciembre del año 2021.



	id_factura	id_cliente	fecha	id_detalle
1	1	9	2021-12-22	1
2	3	4	2021-12-07	3
3	8	0	2021-12-28	8
4	14	6	2021-12-22	14
5	18	7	2021-12-30	18
6	20	8	2021-12-21	20
7	31	12	2021-12-07	31
8	39	26	2021-12-14	39

¿Qué compras han realizado los siguientes clientes?:

- Valentina Anastasia Huerta Corral
- Zayra Manuela Gómez López
- Dante Eduardo Dolores Meza
- Ana Maribel Cedillo Núñez
- Rodrigo Ismael Silva Ugarte.

En este caso escribiremos el siguiente comando:

```
create view cli_view
as select c.nombre,c.apellido, f.id_factura, f.fecha, p.nombre as producto, d.precio
from cliente c
JOIN factura f ON c.id_cliente = f.id_cliente
JOIN detalle d ON f.id_factura = d.id_detalle
JOIN producto p ON d.id_producto = p.id_producto
WHERE c.nombre IN ('Valentina Anastasia', 'Zayra Manuela', 'Dante Eduardo', 'Ana Maribel', 'Rodrigo Ismael')
```

100 %

Messages

Msg 402, Level 16, State 1, Procedure cli\_view, Line 7 [Batch Start Line 0]  
The data types text and varchar are incompatible in the equal to operator.

Completion time: 2025-05-25T01:31:46.4695388-07:00

Este comando nos permitirá crear una vista donde seleccionamos el nombre, apellido de la tabla cliente, id\_factura y fecha de la tabla facturas, nombre del producto de la tabla producto y precio de la tabla detalle por lo que uniremos estos datos utilizando la función join por lo que solo tomaremos el nombre de los clientes, en este caso hay una incompatibilidad de comparar las tablas

de la base de datos ya que están escritos en tipo text y este dato tiene problemas de compatibilidad con los datos de tipo varchar por lo que escribimos lo siguiente para convertir los datos en varchar y poderla comparar

```
create view cli_view
as select CAST(c.nombre AS VARCHAR(MAX)) AS nombre_cliente,
cast (c.apellido as Varchar(max)) as apellido_cliente,
f.id_factura,
f.fecha,
CAST(p.nombre AS VARCHAR(MAX)) AS producto,
d.precio
from cliente c
JOIN factura f ON c.id_cliente = f.id_cliente
JOIN detalle d ON f.id_factura = d.id_detalle
JOIN producto p ON d.id_producto = p.id_producto
WHERE cast (c.nombre as varchar(max)) IN ('Valentina Anastasia', 'Zayra Manuela', 'Dante Eduardo', 'Ana Maribel', 'Rodrigo Ismael')
AND cast(c.apellido as varchar(max)) IN ('Huerta Corral', 'Gómez López', 'Dolores Meza', 'Cedillo Núñez', 'Silva Ugarte');
```

Así que solo agregaremos la función cast para convertir los datos de la base de datos que estén como tipo text a tipo varchar además de los datos después del where donde tenemos los nombres que queremos filtrar y los apellidos.

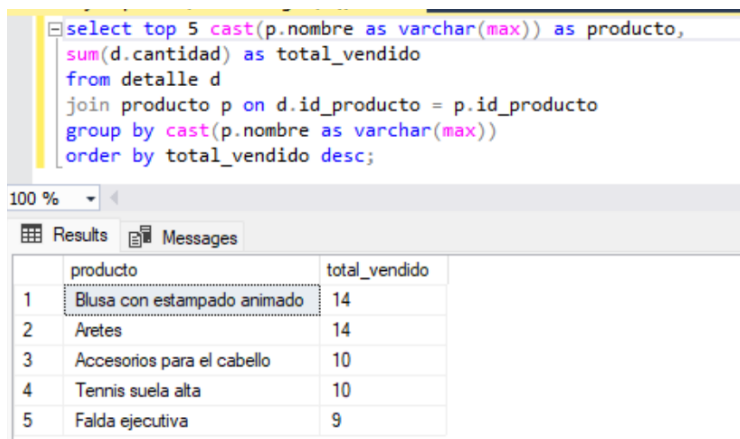
Quiero agregar que es importante verificar los tipos de datos que contiene la base de datos ya que al ser una base de datos importada no me percate de que estaban como tipo texto y me llevo más de una hora y media encontrar el error ya que los comandos no funcionaban, aunque estén bien escritos por la incompatibilidad de los valores por lo que al ejecutar el comando convirtiendo los valores, nos muestra esta tabla que creamos:

select \* from cli\_view

	nombre_cliente	apellido_cliente	id_factura	fecha	producto	precio
1	Ana Maribel	Cedillo Núñez	3	2021-12-07	Aretes	200
2	Ana Maribel	Cedillo Núñez	4	2021-08-24	Chamarra pluma de ganso	110
3	Valentina Anastasia	Huerta Corral	11	2021-07-13	Shorts de Mezclilla	1500
4	Valentina Anastasia	Huerta Corral	12	2022-01-05	Camisón	150
5	Dante Eduardo	Dolores Meza	28	2022-01-22	Suéter cerrado	1200
6	Dante Eduardo	Dolores Meza	33	2022-01-03	Sudadera deportiva	1350
7	Zayra Manuela	Gómez López	34	2021-11-11	Chamarra pluma de ganso	5500

Vemos los clientes que realizaron compras, así como el id de la factura, fecha, producto y su precio, podremos agregar una columna mas con la cantidad para ver cuántos artículos llevo, además de no mostrar al cliente llamado rodrigo ya que probablemente no realizo compras.

Seleccionamos el producto que más ventas ha tenido utilizando el siguiente comando utilizando las conversiones de tipo text a varchar para poder realizar las consultas



```

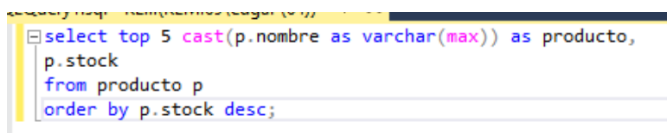
select top 5 cast(p.nombre as varchar(max)) as producto,
sum(d.cantidad) as total_vendido
from detalle d
join producto p on d.id_producto = p.id_producto
group by cast(p.nombre as varchar(max))
order by total_vendido desc;

```

	producto	total_vendido
1	Blusa con estampado animado	14
2	Aretes	14
3	Accesorios para el cabello	10
4	Tennis suela alta	10
5	Falda ejecutiva	9

Seleccionamos el top 5 de los artículos vendidos donde tomaremos el nombre del producto de la tabla producto como tipo varchar además de sumar la cantidad de la tabla detalle en una columna que se llamara total\_vendido por lo que la cantidad la tomaremos de la tabla detalle, unimos el id del producto de la tabla detalle junto con el id del producto de la tabla producto, agrupamos el nombre de los productos permitiéndonos usar la función de suma donde el total vendido será la cantidad de veces que el nombre del producto apareció en la tabla por lo que ordenamos el total vendido de manera descendente pudiendo ver que los artículos de aretes y blusa con estampado animado son los mas vendidos con 14 ventas cada uno.

Verificamos que producto se encuentra con mas stock por lo que realizamos lo siguiente



```

select top 5 cast(p.nombre as varchar(max)) as producto,
p.stock
from producto p
order by p.stock desc;

```

Seleccionamos el top 5 de los productos con mas stock, almacenaremos los datos en

p.stock que vendrá de la tabla productos p, esto lo ordenaremos de manera descendente por lo que nos muestra lo siguiente

	producto	stock
1	Pulseras	78
2	Blusa de flores	70
3	Falda tipo escolar	68
4	Vestido ejecutivo	67
5	Camisa tipo escolar	59

Pulseras es el top 1 de los productos que mas tiene stock en la base de datos.

Ordenamos de la más antigua a la más reciente, las compras que ha habido en la tienda utilizando lo siguiente

```
select f.id_factura, f.fecha,
cast(c.nombre as varchar(max)),
cast(c.apellido as varchar(max)),
cast(p.nombre as varchar(max)) as producto
from factura f
join cliente c on f.id_cliente = c.id_cliente
join producto p on p.id_producto = p.id_producto
order by f.fecha asc;
```

Seleccionamos el id de la factura y fecha de la tabla factura así como el nombre y apellido de la tabla cliente y por último el nombre del producto, relacionamos los datos de la factura con los clientes así como agregamos la tabla de producto mostrándonos lo siguiente:

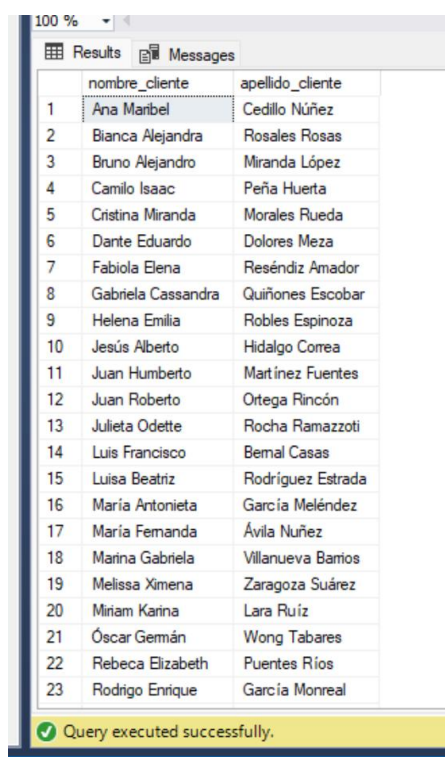
	id_factura	fecha	(No column name)	(No column name)	producto
1	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Blusa de flores
2	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Blusa tipo escocesa
3	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Blusa rayada manga larga
4	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Blusa con estampado animado
5	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Falda larga
6	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Falda tipo escocesa
7	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Falda tipo escolar
8	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Falda ejecutiva
9	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Pantalón acampanado
10	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Skinny Jeans
11	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Pantalón Pants
12	25	2021-06-14	Bianca Alejandra	Rosales Rosas	Shorts de Mezclilla

Donde podemos ver que la fecha más antigua es del 14 de junio del 2021.

Ordenamos alfabéticamente los nombres de todos los clientes de la tienda, para eso realizamos lo siguiente

```
LQuery1.sql - KE...(KEMI69\edgar (64))* X
select cast(nombre as varchar(max)) as nombre_cliente,
cast (apellido as varchar(max)) as apellido_cliente
from cliente
order by nombre_cliente asc, apellido_cliente asc;
```

Seleccionamos los nombres del cliente y apellidos de la tabla cliente donde realizaremos un orden de modo ascendente por lo que vemos la siguiente tabla



	nombre_cliente	apellido_cliente
1	Ana Maribel	Cedillo Núñez
2	Bianca Alejandra	Rosales Rosas
3	Bruno Alejandro	Miranda López
4	Camilo Isaac	Peña Huerta
5	Cristina Miranda	Morales Rueda
6	Dante Eduardo	Dolores Meza
7	Fabiola Elena	Reséndiz Amador
8	Gabriela Cassandra	Quiñones Escobar
9	Helena Emilia	Robles Espinoza
10	Jesús Alberto	Hidalgo Correa
11	Juan Humberto	Martínez Fuentes
12	Juan Roberto	Ortega Rincón
13	Julieta Odette	Rocha Ramazzoti
14	Luis Francisco	Bernal Casas
15	Luisa Beatriz	Rodríguez Estrada
16	María Antonieta	García Meléndez
17	María Fernanda	Ávila Nuñez
18	Marina Gabriela	Villanueva Bamios
19	Melissa Ximena	Zaragoza Suárez
20	Miriam Karina	Lara Ruíz
21	Óscar Germán	Wong Tabares
22	Rebeca Elizabeth	Puentes Ríos
23	Rodrigo Enrique	García Monreal

Query executed successfully.

Ordenado de modo alfabético por nombre del cliente y no por apellido ya que primero ordenamos el nombre y después el apellido.

Seleccionamos cuáles productos pertenecen a cada categoría:

Falda - Pantalón - Chamarra - Zapatos – Accesorios, por lo que realizamos el siguiente código cambiando los datos a varchar, ya que también contiene datos de tipo texto

```

--Query 1.sql - REMI(KEVINOS)edgar (32)
select cast(c.nombre as varchar(max)) as categoria,
cast(p.nombre as varchar(max)) as producto
from producto p
join categoria c on p.id_categoria = c.id_categoria
where cast (c.nombre as varchar(max)) in
('Falda', 'Pantalon', 'Chamarra', 'Zapato', 'Accesorio')
order by categoria, producto;

```

Seleccionamos el nombre de la categoría así como el nombre del producto que vendrán de la tabla producto, uniremos la tabla producto con la tabla de categoría por medio de el id de la categoría, establecemos una selección donde solo tomaremos los datos de chamarra, falda, pantalón, zapato y accesorio y los ordenaremos por categoría y después por producto, por lo que vemos la siguiente tabla:

	categoria	producto
1	Accesorios	Accesorios para el cabello
2	Accesorios	Aretes
3	Accesorios	Collar
4	Accesorios	Pulseras
5	Chamarra	Abrigo largo
6	Chamarra	Cazadora
7	Chamarra	Chamarra delgada
8	Chamarra	Chamarra pluma de ganso
9	Falda	Falda ejecutiva
10	Falda	Falda larga
11	Falda	Falda tipo escocesa
12	Falda	Falda tipo escolar
13	Pantalón	Pantalón acampanado
14	Pantalón	Pantalón Pants
15	Pantalón	Shorts de Mezclilla
16	Pantalón	Skinny Jeans
17	Zapato	Bota alta
18	Zapato	Pantufias
19	Zapato	Tennis suela alta
20	Zapato	Zapato de tacón

Es importante saber que las categorías están escritas de manera correcta ya que si omitimos un acento o lo escribimos en plural, el código no mostrara los datos que estamos filtrando por lo que se necesita tener bien establecidos los nombres de las tablas.

Seleccionamos los encargados de las sucursales de la tienda Akira's Boutique,



En este caso al momento de crear la tabla de sucursal, agregamos una columna con los empleados así como su numero de sucursal por lo que solo seleccionaremos todos los datos de la tabla sucursal utilizando lo siguiente:

```
SQLQuery1.sql - KE...(KEMI69\edgar (52))*
select * from sucursal
```

En este caso no es necesario utilizar la función cast ya que originalmente ingresamos los datos del tipo varchar, por lo que podemos seleccionar todos los datos mostrando lo siguiente

	numsuc	nombre	encargado	direccion	numtelef	ciudad	estado
1	2	Las Mercedes	sonia alejandra fernandez moreno	calle roble #507 fracc. las mercedes	4447831225	san luis potosi	san luis potosi
2	3	obraje	femando calderon ayala	calle dr. jesus diaz de leon #438 col.obraje	4493780921	aguascalientes	aguascalientes
3	4	galerias mazatlan	daniela femanda diaz ordaz	av. de la marina #6204, marina, local 35	6692932059	mazatlan	sinaloa
4	5	zapopan	mario alberto jimenez salcido	av. manuel j. clouthier 525 col. benito juarez	3337841230	zapopan	jalisco
5	6	melchor	yesenia guadalupe campos rojo	av. melchor ocampo #2528 zona centro	6143906721	chihuahua	chihuahua
6	7	constitucion	tamara alejandra bernal ramos	calle constitucion #106 zona centro	6181962954	durango	durango
7	8	centro	samuel enrique bamos encino	av. hidalgo #338 zacatecas centro	4929301250	zacatecas	zacatecas

En esta tabla podremos ver todas las sucursales así como el nombre de la sucursal, dirección y localidad así como el nombre del encargado de la sucursal.

Seleccionamos los empleados que trabajan en la sucursal de Akira's Boutique:

Constitución, para esto revisamos cual es en numero de la sucursal y como es 7, solo tomaremos todos los empleados donde el numero de la sucursal sea 7

```
SQLQuery1.sql - KE...(KEMI69\edgar (52))*
select * from empleados
where sucursal = 7;
```

Por lo que una vez ejecutada este código, muestra lo siguiente:

	id	nombre	apellido_patemo	apellido_matemo	direccion	numtelef	edad	sucursal	correo	contraseña
1	49	tamara alejandra	bernal	ramos	calle del rio colonia colinas	3697854128	28	7	yesenialupita@gmail.com	chetosconqueso3
2	50	ozuna	ojos	claros	calle del cantante 1 colonia de la musica	6325874190	36	7	negritojoclaro25	ozunabeibe
3	51	emili alejandra	uchis	carvero	calle frondosa colonia el frondoso	6547219835	45	7	emycarvero45@gmail.com	carvero5834*
4	52	kendric bryan	lamar	mondragon	terra polar colonia teranova	6324895167	38	7	kendriclmr777@gmail.com	kendry4829
5	53	ann aurora	hattaway	villareal	calle el sendero colonia muy muy lejano	6321548967	29	7	annaurora3846@gmail.com	aurora333*
6	54	diana jizel	lopez	medina	calle 13 colonia las haciendas	6315489243	22	7	jizellpz37@gmail.com	haiku897

Mostrando todos los empleados de la sucursal 7 constitución.

Por ultimo seleccionaremos qué clientes son mayores de 30 años utilizando lo siguiente

```
select * from cliente
where fec_nac < '1995-01-01';
```

Seleccionaremos todos los datos de la tabla cliente donde la fecha de nacimiento sea menor a el primero de enero de 1995 por lo que al estar en el año 2025 todos los datos que se muestren serán clientes con mas de 30 años, mostrando la siguiente tabla

Results		Messages					
	id_cliente	nombre	apellido	direccion	fec_nac	telefono	email
1	0	María Antonieta	García Meléndez	Calle las Girasol #305 col. Jardines	1989-06-15	1828921321	mariantonia@gmail.com
2	1	Óscar Germán	Wong Tabares	Calle Revolución #879 col. México	1994-01-28	2035672143	germanwong@gmail.com
3	2	Valentina Anastasia	Huerta Corral	Calle Flamenco #3493 col. Aviar	1992-07-03	0	valehue92@gmail.com
4	4	Ana Maribel	Cedillo Núñez	Calle Ecuador #278 col. Latinoamérica	1986-09-25	1783920982	anamari86@gmail.com
5	5	Sara Valeria	Jiménez Sánchez	calle Amarillo #890 col. Colores	1990-10-23	1025675109	valesarita@gmail.com
6	6	Rodrigo Enrique	García Monreal	Calle Guadiana #905 fracc. Guadiana	1993-03-29	2146982372	rodrimonreal@gmail.com
7	7	Marina Gabriela	Villanueva Barrios	Calle Luna #560 fracc. Cosmos	1993-06-04	1161728372	marigaby@gmail.com
8	9	Rebeca Elizabeth	Puentes Ríos	Calle Domingo #1929 col. Buendia	1988-10-18	123459834	rebepuentes88@gmail.com
9	10	Yesenia Beatriz	Quintanilla Torres	Calle Roble #405 col. Singapur	1984-01-24	1328902183	betyquintanilla@gmail.com
10	12	Rosa Hortencia	Rosas Morales	Calle Margarita #2892 col Jardines	1983-01-15	1923925489	rositarosas@gmail.com
11	13	Gabriela Cassandra	Quiñones Escobar	Calle Dolores #239 col. Del Río	1987-11-15	2113840192	cassqui87@gmail.com
12	14	Sara Roberta	Moreno Moreno	Calle Colinas #912 col. Colinas del norte	1982-12-14	1244930285	santarober@gmail.com
13	16	Camilo Isaac	Peña Huerta	Calle Niños Héroes #1239 col. México	1987-01-20	1293904389	camisac@gmail.com
14	17	Zayra Manuela	Gómez López	Calle Rosales #390 Fracc. Las flores	1977-10-28	1283094033	zaygom77@gmail.com
15	18	Rosa Alicia	Reyes Nájera	Calle Flautin #1459 col. La Orquesta	1984-11-20	125940940	rosealice84@gmail.com
16	19	Juan Roberto	Ortega Rincón	Calle Los Álamos #4902 Fracc. Monreal	1986-07-31	1489328581	roberortega@gmail.com
17	21	Sandra Isabella	Pereyra Félix	Calle Azalea #901 Fracc. Las Flores	1990-04-09	1289394301	peyfax90@gmail.com
18	22	Rogelia Margarita	Cruz Santos	Calle Golondrinas #9023 col. Felipe Ángeles	1986-05-11	1925678302	margaritacruz@gmail.com
19	23	Julieta Odette	Rocha Ramazzoti	Calle Río Bravo #2013 Fracc. TierraBlanca	1993-07-16	1293391028	julyramazzoti@gmail.com
20	24	Dante Eduardo	Dolores Meza	Calle Miguel Ángel #439 col. Renacimiento	1993-07-15	2023950349	danteduardo93@gmail.com
21	25	Bianca Alejandra	Rosales Rosas	Calle Rueda #1924 col. Emprendedores	1987-06-26	1821039583	rosasbianca@gmail.com
22	26	Fabiola Elena	Reséndiz Amador	Calle Melchor Ocampo #901 col. Libertadores	1994-08-18	1294024901	fabielle94@gmail.com
23	27	Miriam Karina	Lara Ruiz	Sta. María del Oro #423 col. Nazas	1990-07-26	2014893018	minilara@gmail.com

Query executed successfully.

KEMI69\SOLEXPRESS (16.0 RTM) KEMI69\edgar (52) A

Muestra todos los clientes así como su nombre, apellido, dirección, fecha, número de teléfono y correo, cada uno con mas de 30 años cumplidos.



### **Conclusión.**

Aprendimos a realizar consultas del tipo select donde recuperamos datos de diferentes tablas así como realizar filtraciones por medio de el comando where, realizamos uniones de tablas con la función join donde realizamos uniones de diferentes tablas para poder visualizar datos de estas tablas filtrando los datos que queremos ver, así como crear una vista donde también utilizamos las funciones join, aprendimos la importancia de verificar que tipo de datos estaremos buscando ya que hay incompatibilidades de las funciones join con los datos de tipo text por lo que usamos la función cast donde convertiremos el nombre la tabla que buscaremos a tipo varchar con su valor máximo así poder visualizar los datos correctamente, es importante conocer como es que se llaman las tablas que utilizaremos ya que un solo error de gramática hace que el código no funcione correctamente así como la consulta de las categorías de los productos donde los acentos de los productos o los nombres en plural pueden ser consultas erróneas si no revisamos los comandos y los datos correctamente ya que toda esta información puede complicar y hacer mas tardadas las consultas dentro de las bases de datos

### Referencias.

Hackemi. (s. f.-a). *Hackemi/administracion\_BD*. GitHub.

[https://github.com/Hackemi/administracion\\_BD](https://github.com/Hackemi/administracion_BD)

rwestMSFT. (s. f.). = *(Comparación de cadenas o asignación) - SQL Server*. Microsoft

Learn. <https://learn.microsoft.com/es-es/sql/t-sql/language-elements/string-comparison-assignment?view=sql-server-ver16>

Markingmyname. (s. f.). *CAST y CONVERT (Transact-SQL) - SQL server*. Microsoft

Learn. <https://learn.microsoft.com/es-es/sql/t-sql/functions/cast-and-convert-transact-sql?view=sql-server-ver17>