

## Introducción

Este documento tiene la finalidad de exponer y mostrar datos que pueden llegar a ser un interés importante y los cuales podrían en un corto y/o largo plazo ayudar a la empresa **“Taco Riendo”** a crecer a nivel nacional.

### ¿Cómo se realizó la obtención de Datos?

La obtención de estos se realizó mediante al acceso de la Base de Datos de **“Taco Riendo”** la cual se creó mediante el Sistema de Gestión de Bases de Datos **PostgreSQL**, de igual manera para facilitar el trabajo y que fuera más conciso se hizo uso del Software **pgAdmin4**.

## Consultas realizadas

A continuación se muestran las consultadas realizadas por el equipo de desarrolladores de **Computólogos A.C..**

### • LOS 15 CLIENTES QUE MÁS FRECUENTAN LA FRANQUICIA

```
1      SELECT
2          CURP , Nombre , A_Paterno , count(CURP) "Total de compras" , SUM(Total)
3      "Gastos totales"
4      FROM
5          cliente
6      INNER JOIN
7          ticket
8      ON
9          cliente.CURP=ticket.id_Cliente
10     GROUP BY
11         cliente.CURP
12     ORDER BY
13         4 DESC
14     LIMIT
15         15;
```

	curp [PK] character varying (18)	nombre character varying (50)	a_paterno character varying (50)	Total de compras bigint	Gastos totales numeric
1	VJON479936KSQXFIV7	Carolina	Flores	25	14241.27
2	HXZL562333KILPXR54	Raul	Ruiz	23	12077.92
3	EMOH844555BXNQSG65	Samuel	Diaz	21	9753.03
4	UWJB054687VZEHVQY1	Inmaculada	Lozano	20	9973.33
5	RDIJ180491AWTDNC98	Noelia	Crespo	20	12183.63
6	VAFB273357HIUZXQE0	Diego	Rodriguez	19	10990.03
7	LCAP931156YUSFAL58	Jordi	Gonzalez	19	9921.65
8	RELX927057QWXZNY88	Fatima	Ortiz	19	8928.26
9	JUGO699555BBBLRPU0	Josefa	Ramos	19	10208.48
10	CBTG326403LYENTQ09	Mateo	Herrero	19	8831.68
11	ZHFO102076JDLXPVS8	Marco	Carrasco	19	10584.48
12	CWWQ453959YJBHXA30	Ines	Moreno	19	9174.84
13	HTGP251127GBLHZHJ7	Xavier	Perez	19	7787.96
14	MXSS365254ECUMDP16	Javier	Gallego	19	9051.56
15	MNAZ768626UIIFCHG4	Monica	Vazquez	19	8030.10

Con esta consulta se pensó en que podría surgir la idea de recompensar a estos clientes por lealtad a la franquicia.

## • INGRESOS TOTALES DE LAS TAQUERIAS

```

1      SELECT id_taqueria, Nombre, SUM(Total) "Ingresos totales"
2      FROM ticket INNER JOIN taqueria
3          ON ticket.id_taqueria = taqueria.id
4          GROUP BY id_Taqueria, Nombre
5          ORDER BY 1;
6

```

<b>Id_taqueria</b>	<b>nombre</b>	<b>Ingresos totales</b>
integer	character varying (100)	numeric
1	Sheelah	246440.63
2	Willard	291455.93
3	Christiana	260541.05
4	Melloney	267506.75
5	Shandy	249686.97
6	Hilliard	241346.12
7	Tori	255861.70
8	Cherri	269421.20
9	Zebulen	283466.80
10	Dee	250112.61
11	Jan	286765.48
12	Carlen	261584.33
13	Vick	243801.78
14	Kacy	257016.16
15	Latachia	254894.50
16	Enrico	256124.03
17	Gaby	250214.66
18	Burty	270943.78
19	Sibyl	277265.61
20	Lorilyn	264910.90

Con esta consulta los dueños pueden tener una imagen a gran escala de cuanto ha generado cada una de las sucursales a lo largo de su existencia.

#### • EMPLEADOS QUE CUMPLEN CON EL PROGRAMA DE ANTIGÜEDAD

```

1      SELECT CURP, RFC, id_Taqueria, Nombre, A_Paterno, Antiguedad FROM
2      taquero
3          WHERE DATE_PART('year', AGE(NOW(), antiguedad)) >= 2
4          UNION
5          SELECT CURP, RFC, id_Taqueria, Nombre, A_Paterno, Antiguedad FROM
6      parrillero
7          WHERE DATE_PART('year', AGE(NOW(), antiguedad)) >= 2
8          UNION
9          SELECT CURP, RFC, id_Taqueria, Nombre, A_Paterno, Antiguedad FROM
10     mesero
11        WHERE DATE_PART('year', AGE(NOW(), antiguedad)) >= 2
12        UNION
13        SELECT CURP, RFC, id_Taqueria, Nombre, A_Paterno, Antiguedad FROM
14     cajero
15        WHERE DATE_PART('year', AGE(NOW(), antiguedad)) >= 2
16        UNION
17        SELECT CURP, RFC, id_Taqueria, Nombre, A_Paterno, Antiguedad FROM
18     tortillero
19        WHERE DATE_PART('year', AGE(NOW(), antiguedad)) >= 2
20        UNION
21        SELECT CURP, RFC, id_Taqueria, Nombre, A_Paterno, Antiguedad FROM
22     repartidor
23        GROUP BY CURP
24        HAVING DATE_PART('year', AGE(NOW(), antiguedad)) >= 2;

```

	curp character varying (18)	rfc character varying (13)	id_taqueria integer	nombre character varying (50)	a_paterno character varying (50)	antiguedad date
1	VZCR961550YXIEVRU1	UUZI814855097	78	Fatima	Soto	2019-04-08
2	AVCI795557YMFSLV1	XXTY524720RVV	45	Luisa	Aguilar	2019-12-25
3	WTJV386328XKZERN45	FVJD5709433BA	78	Daniel	Sanz	2019-03-14
4	MYNA347248VLFJKFN6	DGSI355474PDK	83	Santiago	Nieto	2019-06-29
5	JFIS292245ZJEDAMG9	AFMU528735DF1	57	Jesus	Gonzalez	2018-10-03
6	QTLK941752TJFUKOW3	NQMF273542BB2	95	Hugo	Sanchez	2019-11-11
7	UVRE628915UBPCBWA4	JIHQ4499323I0	68	Amparo	Jimenez	2020-02-02
8	FDAH564090APZBZGU1	JPHG651684506	6	Anna	Martin	2019-07-28
9	VCSJ042947KQDBVB05	WQTG349357525	56	Cristian	Pascual	2019-12-21
10	SRLZ059734MRYJQVG5	LLYL098237G1S	32	Cristian	Rubio	2020-03-14
11	XSFY624645YMRCVFE4	AGMS356489NPH	44	Miriam	Moya	2019-03-13
12	YSJH020532XCJZLOF9	BMNR181181CD1	22	Adrian	Santos	2019-06-05
13	SIRV756370DHSLOCS7	BWUM948934TG5	15	Aurora	Gil	2019-03-25
14	FBJY551249WHSEJKY0	AHRH664798T0N	83	Andrea	Delgado	2019-05-13
15	JEZB075776IRARTV09	WUJX966820870	3	Diego	Montero	2020-02-07
16	JWIF381053PBNXVJ0	IVCF380407XY5	61	Luis	Castillo	2020-04-01
17	UKBV143405FQDNVUN9	FOLW5796230W6	32	Yolanda	Gimenez	2019-07-18
18	NZSE879897JMJKCUQ7	YSLC1957149HW	94	Julio	Iglesias	2020-04-22
19	ZIPL429136VHYLQAY5	MJPZ245972XJN	89	Raquel	Carmona	2020-02-26
20	CIDJ736695ALDXGUG2	EHOK208662DRA	10	German	Gutierrez	2019-08-23

A partir podemos obtener todos los trabajadores, sin importar el puesto en el que se encuentren, los cuales cumplen con el requisito para ser aptos para el programa de Antigüedad.

- **NUMERO DE CLIENTES ATENDIDOS POR SUCURSAL**

```

1
2      SELECT id_taqueria, Nombre, COUNT(id_taqueria) "Clientes atendidos
totales"
3          FROM ticket INNER JOIN taqueria
4              ON ticket.id_taqueria = taqueria.id
5              GROUP BY id_taqueria, Nombre
6              ORDER BY 3 DESC;

```

	<b>Id_taqueria</b> integer	<b>nombre</b> character varying (100)	<b>Clientes atendidos totales</b> bigint
1	2	Willard	557
2	11	Jan	545
3	100	Ogdon	540
4	99	Demetris	539
5	51	Melinde	538
6	19	Sibyl	537
7	74	Lukas	530
8	95	Arleta	529
9	9	Zebulen	529
10	39	Constancia	529
11	29	Irving	528
12	42	Ralph	526
13	53	Darryl	525
14	60	Arvy	524
15	18	Burty	524
16	88	Ammamaria	523
17	81	Zechariah	522
18	8	Cherri	521
19	61	Dena	521
20	79	Tait	519

Esta consulta tiene como fin saber cuantos gentes se han atendido en cada sucursal a lo largo de su existencia, lo cual podría ayudar a saber si se debe contratar más personal.

- **LOS 20 MESEROS CON MÁS CLIENTES ATENDIDOS**

```

1   SELECT RFC, Nombre, A_Paterno, A_Materno, Telefono, Email, COUNT(
2     RFC_mesero) "Clientes atendidos"
3     FROM mesero INNER JOIN ticket ON mesero.RFC = ticket.RFC_mesero
4     GROUP BY RFC, Nombre, A_Paterno, A_Materno, Telefono, Email
5     ORDER BY 7 DESC
6     LIMIT 20;

```

	rfc character varying (13)	nombre character varying (50)	a_paterno character varying (50)	a_materno character varying (50)	telefono character varying (10)	email character varying (320)	Clientes atendidos bigint
1	KAQW7893590KI	Miguel	Benitez	Cortes	6248471057	scullinan3t@google.nl	71
2	JLQK9639762FU	Eva	Ferrer	Fuentes	9848326071	gsawdyb@cnn.com	67
3	ACVM887096TKD	Hector	Vazquez	Gomez	4058729385	rfroggatp3@nationalgeographic.com	65
4	MTBM880091YPC	Diana	Santos	Ruiz	5046231329	fmalthouse4@liveinternet.ru	65
5	XBHR841850KQQ	Celia	Medina	Fernandez	7612304264	fsissensy@deliciousdays.com	65
6	MOMZ248703D93	Victor	Suarez	Vargas	9864974591	wbonallackmo@stumbleupon.com	64
7	HOCK149446RYT	Jaime	Flores	Montez	9197788313	tpenningq@epa.gov	64
8	PGWH495792K8W	Fatima	Uriostegui	Diaz	2461281518	povingtonj@freewebs.com	64
9	FOLW5796230W6	Yolanda	Gimenez	Gallardo	2989281203	kgreganqh@arizona.edu	64
10	PQRV457489EP4	Ivan	Guerrero	Crespo	7622291660	lpirdy7z@bbc.co.uk	64
11	NIXY203235AXD	Angel	Ortiz	Benitez	9471938712	enicholls4k@163.com	63
12	KRNU675338XQW	Ramon	Herrera	Alonso	5797864264	niorizzq8@digio.com	63
13	HBVC277823J4K	Esther	Sanz	Diez	3678468169	cwillingamkr@bbc.co.uk	63
14	KVED52225099Q	Alejandro	Mora	Rubio	8861270997	kschrei4a@redcross.org	63
15	NHGW401102KCL	Monica	Reyes	Hidalgo	5924313871	wslocombc4@myspace.com	63
16	APYD112423IDU	Cristobal	Calvo	Molina	4855517123	gdowdf@imageshack.us	62
17	QFBN041308GWA	Manuela	Vicente	Garcia	4009216753	mlerelj@list-manage.com	62
18	TEOI443805GL0	Lucia	Gonzalez	Diaz	1849245137	bwastielw@a8.net	62
19	AQDG777687Q9V	Aurora	Gallego	Parra	5482198908	tcooteega@nbcnews.com	62
20	BAWY057841YXL	Rosario	Arias	Parra	9911571705	jbruno73@ihg.com	62

A partir de los datos que proporciona esta consulta, se podría dar un bono a estos meseros por su compromiso laboral.

- LAS 10 TAQUERIAS QUE HAN TENIDO QUE ABASTECERSE MÁS

```

1   SELECT id_taqueria, nombre, SUM(PrecioCompra) "Gastos totales en
2   abastencimiento"
3   FROM abastecer
4   INNER JOIN compra ON compra.id = abastecer.id_Compra
5   INNER JOIN taqueria ON abastecer.id_Taqueria = taqueria.id
6   GROUP BY id_taqueria, nombre
7   ORDER BY 3 DESC
    LIMIT 10;

```

	<b>id_taqueria</b> integer	<b>nombre</b> character varying (100)	<b>Gastos totales en abastecimiento</b> numeric
1	27	Cherida	102543.05
2	40	Dusty	82962.34
3	77	Julee	72032.61
4	2	Willard	70998.38
5	3	Christiana	66327.52
6	78	Brockie	62756.29
7	90	Hirsch	62645.86
8	51	Melinde	60002.28
9	18	Burty	52417.06
10	94	Marti	51779.90

Con estos datos se podría tener una mejor idea de cuánto debe gastar estas sucursales en abastecimiento de productos en el futuro.

- TAQUERIAS VENDEN MÁS POR PUNTOS ”TAQUERO CORAZÓN”

```

1   SELECT id_taqueria, Nombre, COUNT(id_taqueria) "Pagos con Puntos Taquero
Corazon"
2     FROM ticket INNER JOIN taqueria
3       ON ticket.id_taqueria = taqueria.id
4     GROUP BY id_taqueria, Nombre, Es_Puntos
5     HAVING Es_Puntos = true
6     ORDER BY 3 DESC;

```

	<b>Id_taqueria</b> integer	<b>nombre</b> character varying (100)	<b>Pagos con Puntos Taquero Corazon</b> bigint
1	37	Cristal	189
2	9	Zebulen	189
3	2	Willard	188
4	99	Demetris	186
5	67	Felizio	186
6	13	Vick	186
7	93	Edyth	185
8	11	Jan	185
9	95	Arleta	185
10	19	Sibyl	182
11	59	Fidole	182
12	70	Magdalen	181
13	66	Gwennie	180
14	82	Shirley	180
15	46	Tammi	179
16	75	Brendan	179
17	18	Burty	179
18	96	Prue	178
19	100	Ogdon	178
20	87	Elvina	177

De estos datos obtenidos se puede saber en que sucursales se usan más los puntos **“TAQUERO CORAZÓN”** lo cual nos puede dar un indicio de que sucursales generan más ingresos.

- **INGREDIENTES DE SALSAS CON MAXIMO PICOR**

```

1      SELECT
2          id_salsa,
3              sls.nombre,
4                  array_to_string(array_agg(array[ing.nombre]), ', ') AS
5                      ingredientes
6                          FROM
7                              utilizar_en_salsa AS us
8                          JOIN
9                              ingrediente AS ing
10                         ON us.id_ingrediente = ing.id
11                         JOIN
12                             salsa AS sls
13                             ON us.id_salsa = sls.id
14                         WHERE
15                             sls.picor='Extremo'
16                         GROUP BY
17                             us.id_salsa, sls.nombre
18                         ORDER BY
19                             us.id_salsa ASC;

```

	<b>id_salsa</b> integer	<b>nombre</b> character varying (50)	<b>ingredientes</b> text
1	4	Salsa roja	Chile - Jalapeno, Ajos, Cilantro, Cebolla, Jitomates
2	5	Salsa de chile habanero	Chile - Habanero, Ajos, Cilantro, Cebolla, Jitomates, Tomates
3	7	Salsa chipotle	Chile - Chipotle, Ajos, Cilantro, Cebolla, Jitomates
4	11	Salsa de molcajete	Chile - Jalapeno, Ajos, Cilantro, Cebolla, Tomates

Con esta consulta se pretende que pueda ser utilizada por los empleados al momento de querer preparar las salsas con el picor más grande.

- INGRESO DE TODAS LAS SUCURSALES EN EL ÚLTIMO MES

```

1      SELECT id_taqueria, Nombre, SUM(Total) "Ingresos en el ltimo mes"
2          FROM (
3              SELECT
4                  id_taqueria, Nombre , Total
5                      FROM
6                          ticket INNER JOIN taqueria ON ticket.id_taqueria = taqueria.id
7                          GROUP BY id_taqueria, Nombre, Fecha, Total
8                          HAVING
9                              Fecha >= '2022-05-15' AND Fecha <= '2022-06-15'
10                         ) AS tk
11                         GROUP BY
12                             id_taqueria, Nombre
13                             ORDER BY 3 DESC;

```

	<b>Id_taqueria</b> integer	<b>nombre</b> character varying (100)	<b>Ingresos en el último mes</b> numeric
1	77	Julee	11544.21
2	37	Cristal	9881.60
3	10	Dee	8851.74
4	22	Alverta	8765.39
5	40	Dusty	8626.30
6	28	Ulla	8579.34
7	87	Elvina	8570.05
8	76	Augustin	8555.25
9	57	Elene	8205.40
10	5	Shandy	7854.29
11	30	Dmitri	7642.31
12	14	Kacy	7522.14
13	20	Lorilyn	7443.07
14	39	Constancia	7367.62
15	47	Trixie	7361.63
16	55	Chester	7315.31
17	70	Magdalen	7293.40
18	75	Brendan	7221.73
19	11	Jan	7168.34
20	31	Lisetta	7137.73

Con estos datos los dueños pueden tener una perspectiva más amplia del ingreso por mes de cada sucursal.

- **PLATILLOS DE CARNITAS VENDIDOS EN TODAS LAS SUCURSALES**

```

1   SELECT
2       id,nombre ,pv.piezas_vendidas ,precio
3   FROM
4       item
5   JOIN
6       (SELECT id_item,SUM(cantidad) AS piezas_vendidas
7        FROM vender GROUP BY id_item ORDER BY id_item ASC) AS pv
8   ON
9       id=pv.id_item
10      GROUP BY
11          item.id,pv.piezas_vendidas
12      HAVING
13          nombre LIKE '%carnitas%'
14      ORDER BY item.id ASC;

```

	<b>Id</b> [PK] integer	<b>nombre</b> character varying (50)	<b>piezas_vendidas</b> bigint	<b>precio</b> numeric
1	1	Taco de carnitas	814	14
2	18	Burrito de carnitas	732	30
3	35	Quesadilla de carnitas	786	27
4	52	Gringa de carnitas	850	18
5	69	Torta de carnitas	708	28

Con esta consulta se pretende saber cuantas piezas de todos los productos que llevan carnitas se han vendido.

- CERVEZA MODELO VENDIDA POR TAQUERIA EN EL ULTIMO MES

```

1      SELECT
2          id_taqueria , id_item , SUM(cantidad) AS cerveza_vendida
3      FROM (
4          SELECT
5              id_taqueria , id_item , cantidad
6          FROM
7              ticket
8          JOIN
9              (SELECT * FROM vender WHERE id_item=111) AS ventas
10         ON
11             id=ventas.id_ticket
12         GROUP BY
13             id_taqueria , id_item , fecha , cantidad
14         HAVING
15             fecha >= ('now'::timestamp - '1 month'::interval)
16     ) AS foo
17     GROUP BY
18         id_taqueria , id_item
19     ORDER BY
20         id_taqueria ASC;

```

	<b>id_taqueria</b> integer	<b>id_item</b> integer	<b>cerveza_vendida</b> bigint
1	2	111	14

Esta consulta es parecida a la anterior, la intención es saber cuantas cervezas **Modelo** se han vendido por sucursal.

- TACOS AL PASTOR VENDIDOS LOS DIAS LUNES DE ESTE ÚLTIMO MES POR SUCURSAL

```

1      SELECT id_taqueria, nombre, SUM(cantidad) "Tacos al pastor vendidos
2          los lunes del ltimo mes"
3      FROM
4          (SELECT id_taqueria,id_item, id_ticket, cantidad
5              FROM
6                  vender JOIN ticket ON vender.id_ticket = ticket.id
7                  GROUP BY id_taqueria,id_item,id_ticket,Fecha
8                  HAVING Fecha >= '2022-05-15' AND Fecha <= '2022-06-15' AND EXTRACT
9                      (dow from Fecha) = 1 AND id_item = 2
10                 ) as tv
11                 INNER JOIN
12                     taqueria
13                     ON
14                         tv.id_taqueria = taqueria.id
15
16                     GROUP BY id_taqueria, nombre
17
18                     ORDER BY 2 DESC;

```

	<b>id_taqueria</b> integer	<b>nombre</b> character varying (100)	<b>Tacos al pastor vendidos los lunes del último mes</b> bigint	
1	37	Cristal	35	

Con estos datos se pretende saber si la promoción “**Lunes de 2x1 en tacos al pastor**” realmente genera los ingresos que se esperaría que haga.

- **LOS 3 TACOS MÁS VENDIDOS**

```

1      SELECT
2          id,nombre ,pv.piezas_vendidas
3      FROM
4          item
5      JOIN (
6          SELECT
7              id_item ,SUM(cantidad) AS piezas_vendidas
8          FROM
9              vender
10             GROUP BY id_item
11         ) AS pv
12         ON
13             id=pv.id_item
14             GROUP BY
15                 item.id,pv.piezas_vendidas
16             HAVING
17                 nombre LIKE '%Taco%'
18             ORDER BY pv.piezas_vendidas DESC LIMIT 3;

```

	<b>Id</b> [PK] integer	<b>nombre</b> character varying (50)	<b>piezas_vendidas</b> bigint
1	2	Taco al pastor	924
2	4	Taco de suadero	858
3	11	Taco de tuétano	847

Con esta consulta, se tiene como fin saber cuales son los tacos más vendidos con el fin de saber si se deben abastecer más de los ingredientes que llevan y generar mayores ingresos.

- **TOP 10 PRODUCTOS MÁS VENDIDOS EN LO QUE VA DEL AÑO**

```

1   SELECT id_item, nombre, SUM(cantidad) "Cantidad de piezas vendidas
2   en el transcurso de año"
3   FROM
4       (SELECT id_item, cantidad
5        FROM
6            vender JOIN ticket ON vender.id_ticket = ticket.id
7            GROUP BY id_item, cantidad, Fecha
8            HAVING Fecha >= '2022-01-01' AND Fecha <= NOW()
9        ) as tv
10      JOIN
11          item
12      ON
13          tv.id_item = item.id
14      GROUP BY id_item, nombre
15      ORDER BY 3 DESC
16      LIMIT 10;
17

```

	<b>id_item</b> integer	<b>nombre</b> character varying (50)	<b>Cantidad de piezas vendidas en el transcurso de año</b> bigint
1	93	Cochinita pibil	217
2	96	Agua de jamaica	177
3	4	Taco de suadero	176
4	1	Taco de carnitas	165
5	90	Enchiladas	165
6	28	Burrito de tuétano	151
7	97	Aqua de tamarindo	149
8	84	Torta de chorizo	149
9	58	Gringa de cecina	144
10	111	Modelo	141

A partir de estos datos, podemos saber que productos son los más solicitados en las sucursales lo cual puede dar una idea de que productos producir más.

- TACOS MAS VENDIDOS EN CIERTA SUCURSAL

```

1      SELECT
2          id_taqueria,nombre,SUM(cantidad) AS total
3      FROM
4          ticket
5      JOIN
6          (SELECT id_ticket,id_item,nombre,cantidad FROM vender JOIN item
7      as it ON id_item=it.id GROUP BY id_ticket,id_item,nombre HAVING nombre LIKE '%Taco%'
8          ORDER BY cantidad DESC)
9      AS foo
10     ON ticket.id = foo.id_ticket
11     WHERE id_taqueria=1
12     GROUP BY id_taqueria,nombre
13     ORDER BY total DESC;

```

	<b>id_taqueria</b> integer	<b>nombre</b> character varying (50)	<b>total</b> bigint
1	1	Taco de bistec	43
2	1	Taco de chorizo	24
3	1	Taco de cochinita pibil	24
4	1	Taco al pastor	21
5	1	Taco de lengua	20
6	1	Taco de chapulines	20
7	1	Taco de chilorio	16
8	1	Taco de requesón	14
9	1	Taco de cecina	13
10	1	Taco de barbacoa	10

Con esta consulta, podemos ver que tacos se venden más por una sucursal específica, en este caso la sucursal con *id* = 1, pero podemos modificar este parámetro.

- TACOS MAS VENDIDOS POR SUCURSAL

```

1      SELECT
2          id_taqueria , nombre , SUM(cantidad) AS total
3      FROM
4          ticket
5      JOIN
6          (SELECT id_ticket , id_item , nombre , cantidad FROM vender JOIN item
7           as it ON id_item=it.id GROUP BY id_ticket , id_item , nombre HAVING nombre LIKE ,
8           '%Taco%')
9          ORDER BY cantidad DESC)
10         AS foo
11     ON ticket.id = foo.id_ticket
12     GROUP BY id_taqueria , nombre
13     ORDER BY id_taqueria ASC , total DESC;

```

	<b>id_taqueria</b> integer	<b>nombre</b> character varying (50)	<b>total</b> bigint
1	1	Taco de bistec	43
2	1	Taco de chorizo	24
3	1	Taco de cochinita pibil	24
4	1	Taco al pastor	21
5	1	Taco de chapulines	20
6	1	Taco de lengua	20
7	1	Taco de chilorio	16
8	1	Taco de requesón	14
9	1	Taco de cecina	13
10	1	Taco de barbacoa	10
11	2	Taco de requesón	22
12	2	Taco de suadero	19
13	2	Taco de tuétano	19
14	2	Taco de chorizo	19
15	2	Taco de lengua	15
16	2	Taco de carnitas	5
17	2	Taco de tripas	5
18	2	Taco de nopal asado	2
19	2	Taco de cabeza	2
20	2	Taco al pastor	2
21	3	Taco de cecina	32
22	3	Taco de lengua	26
23	3	Taco de bistec	18

Con esta consulta, podemos ver todos los tipos de tacos que se han vendido por cada sucursal, es parecida a la anterior consulta, pero aquí se lista la información de todas las sucursales.

## Conclusiones

Podemos observar que el sistema de la empresa permite realizar la obtención de bastante información interesante e importante, la cual podrá servir para realizar diferentes decisiones y estrategias de negocios las que beneficiarán de manera impactante a cada una de las sucursales existentes.