National University

Of Computer & Emerging Sciences

Mamoon Khan

20F – 0361

BCS –7A

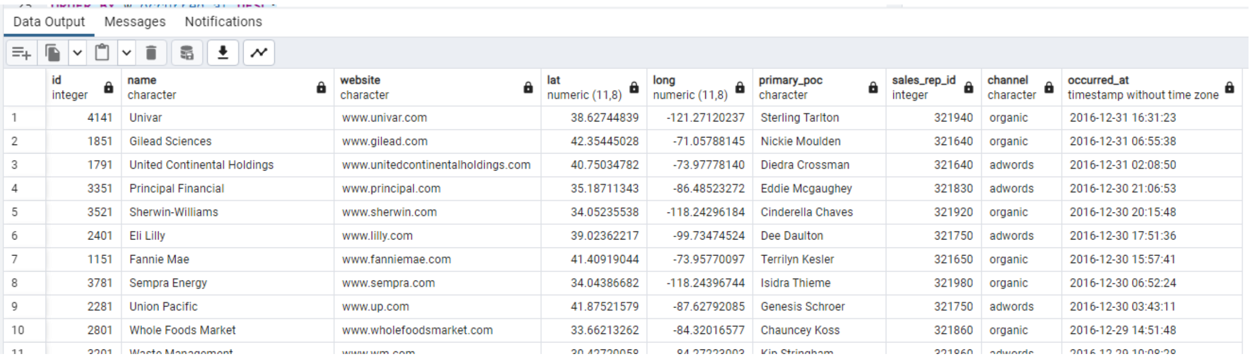
Data Science Assignment-03

QUEST`ION 1

CODE:

```
SELECT a.id,
            a.name,
            a.website,
            a.lat,
            a.long,
            a.primary_poc,
            a.sales_rep_id,
            w.channel,
            w.occurred_at
FROM accounts a
INNER JOIN web_events w
ON a.id = w.account_id
WHERE (w.channel = 'organic' OR w.channel = 'adwords')
            AND EXTRACT(YEAR FROM w.occurred_at) = '2016'
ORDER BY w.occurred_at DESC;
```

SCREEN SHOT:



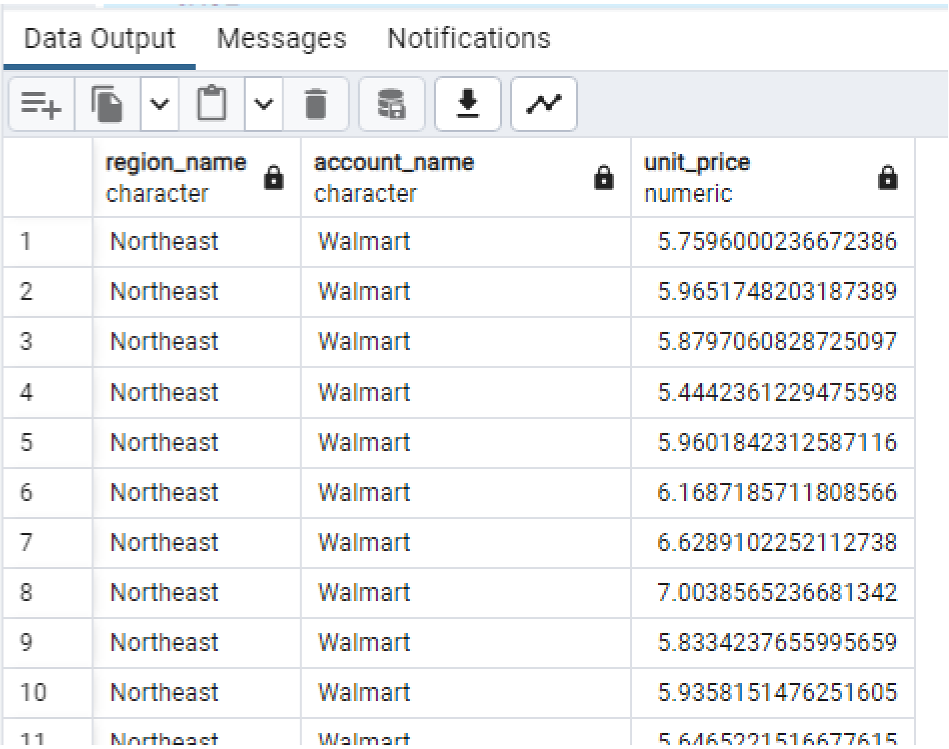
	id integer	name character	website character	lat numeric (11,8)	long numeric (11,8)	primary_poc character	sales_rep_id integer	channel character	occurred_at timestamp without time zone
1	4141	Univar	www.univar.com	38.62744839	-121.27120237	Sterling Tarlton	321940	organic	2016-12-31 16:31:23
2	1851	Gilead Sciences	www.gilead.com	42.35445028	-71.05788145	Nickie Moulden	321640	organic	2016-12-31 06:55:38
3	1791	United Continental Holdings	www.unitedcontinentalholdings.com	40.75034782	-73.97778140	Diedra Crossman	321640	adwords	2016-12-31 02:08:50
4	3351	Principal Financial	www.principal.com	35.18711343	-86.48523272	Eddie Mcgaughey	321830	adwords	2016-12-30 21:06:53
5	3521	Sherwin-Williams	www.sherwin.com	34.05235538	-118.24296184	Cinderella Chaves	321920	organic	2016-12-30 20:15:48
6	2401	Eli Lilly	www.lilly.com	39.02362217	-99.73474524	Dee Daulton	321750	adwords	2016-12-30 17:51:36
7	1151	Fannie Mae	www.fanniemae.com	41.40919044	-73.95770097	Terrilyn Kesler	321650	organic	2016-12-30 15:57:41
8	3781	Sempra Energy	www.sempra.com	34.04386682	-118.24396744	Isidra Thieme	321980	organic	2016-12-30 06:52:24
9	2281	Union Pacific	www.up.com	41.87521579	-87.62792085	Genesis Schroer	321750	adwords	2016-12-30 03:43:11
10	2801	Whole Foods Market	www.wholefoodsmarket.com	33.66213262	-84.32016577	Chauncey Koss	321860	organic	2016-12-29 14:51:48
11	3201	Waste Management	www.wm.com	30.42720058	-84.27223003	Kip Stringham	321860	adwords	2016-12-29 10:08:28

QUESTION 2:

CODE:

```
SELECT r.name AS region_name,
        a.name AS account_name,
   CASE
              WHEN o.total_amt_usd = 0 THEN 0
              ELSE o.total_amt_usd / (o.total + 0.01)
              END AS unit_price
FROM orders o
INNER JOIN accounts a
ON o.account_id = a.id
INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
INNER JOIN region r
ON s.region_id = r.id;
```

SCREEN SHOT:



	region_name character	account_name character	unit_price numeric
1	Northeast	Walmart	5.7596000236672386
2	Northeast	Walmart	5.9651748203187389
3	Northeast	Walmart	5.8797060828725097
4	Northeast	Walmart	5.4442361229475598
5	Northeast	Walmart	5.9601842312587116
6	Northeast	Walmart	6.1687185711808566
7	Northeast	Walmart	6.6289102252112738
8	Northeast	Walmart	7.0038565236681342
9	Northeast	Walmart	5.8334237655995659
10	Northeast	Walmart	5.9358151476251605
11	Northeast	Walmart	5.6465221516677615

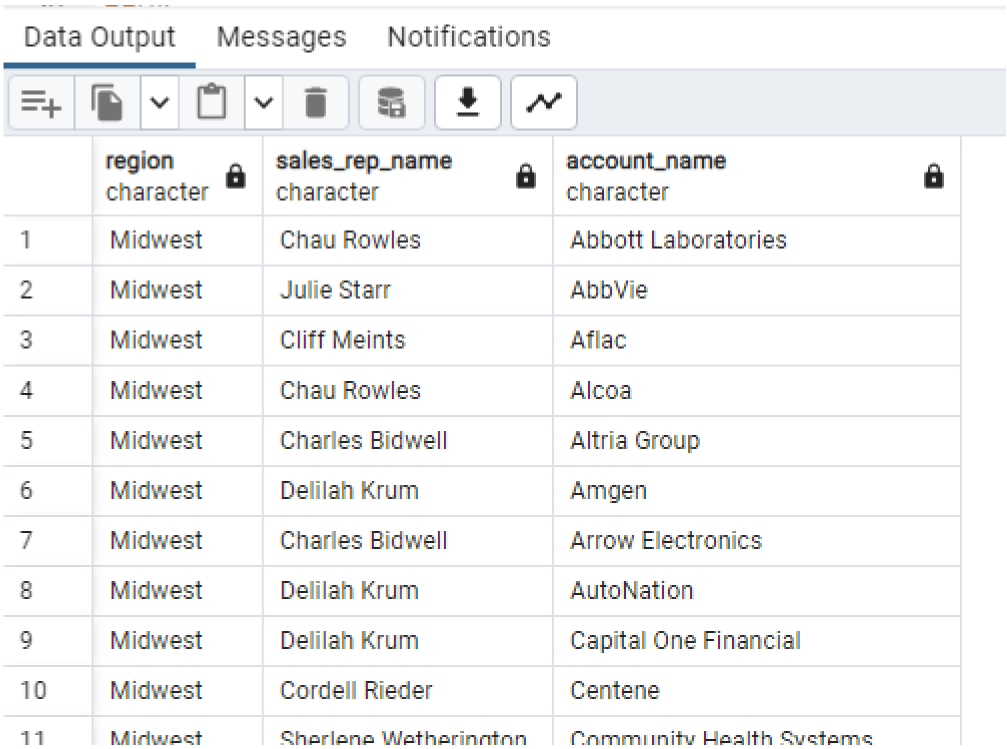
QUESTION 3

CODE:

```
SELECT r.name as region,
        s.name as sales_rep_name,
        a.name as account_name
FROM
accounts a INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
INNER JOIN region r
ON s.region_id = r.id
WHERE
        r.name='Midwest'
```

```
ORDER BY account_name ASC;
```

SCREEN SHOT:



	region character	sales_rep_name character	account_name character
1	Midwest	Chau Rowles	Abbott Laboratories
2	Midwest	Julie Starr	AbbVie
3	Midwest	Cliff Meints	Aflac
4	Midwest	Chau Rowles	Alcoa
5	Midwest	Charles Bidwell	Altria Group
6	Midwest	Delilah Krum	Amgen
7	Midwest	Charles Bidwell	Arrow Electronics
8	Midwest	Delilah Krum	AutoNation
9	Midwest	Delilah Krum	Capital One Financial
10	Midwest	Cordell Rieder	Centene
11	Midwest	Sherlene Wetherington	Community Health Systems

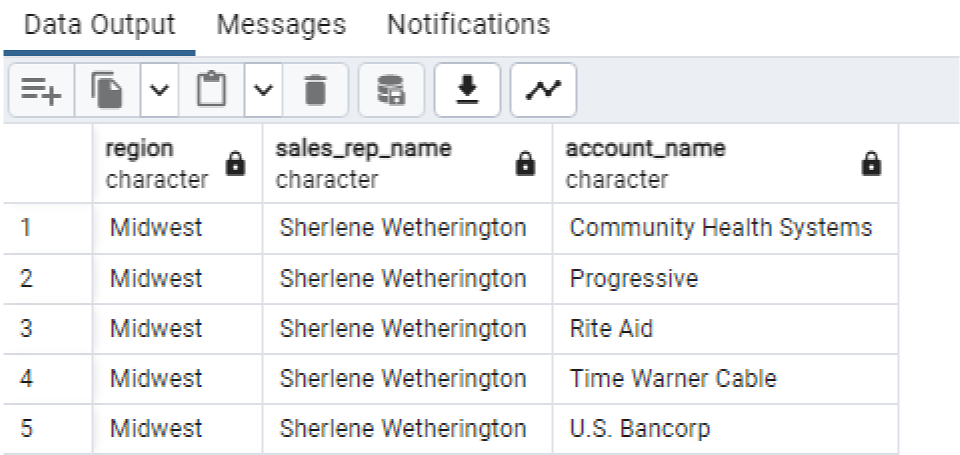
QUESTION 4

CODE:

```
SELECT r.name as region,
            s.name as sales_rep_name,
            a.name as account_name
FROM
accounts a INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
```

```
INNER JOIN region r
ON s.region_id = r.id
WHERE
        (s.name LIKE 'S%' AND r.name='Midwest')
ORDER BY account_name ASC;
```

SCREEN SHOT:



	region	sales_rep_name	account_name
	character	character	character
1	Midwest	Sherlene Wetherington	Community Health Systems
2	Midwest	Sherlene Wetherington	Progressive
3	Midwest	Sherlene Wetherington	Rite Aid
4	Midwest	Sherlene Wetherington	Time Warner Cable
5	Midwest	Sherlene Wetherington	U.S. Bancorp

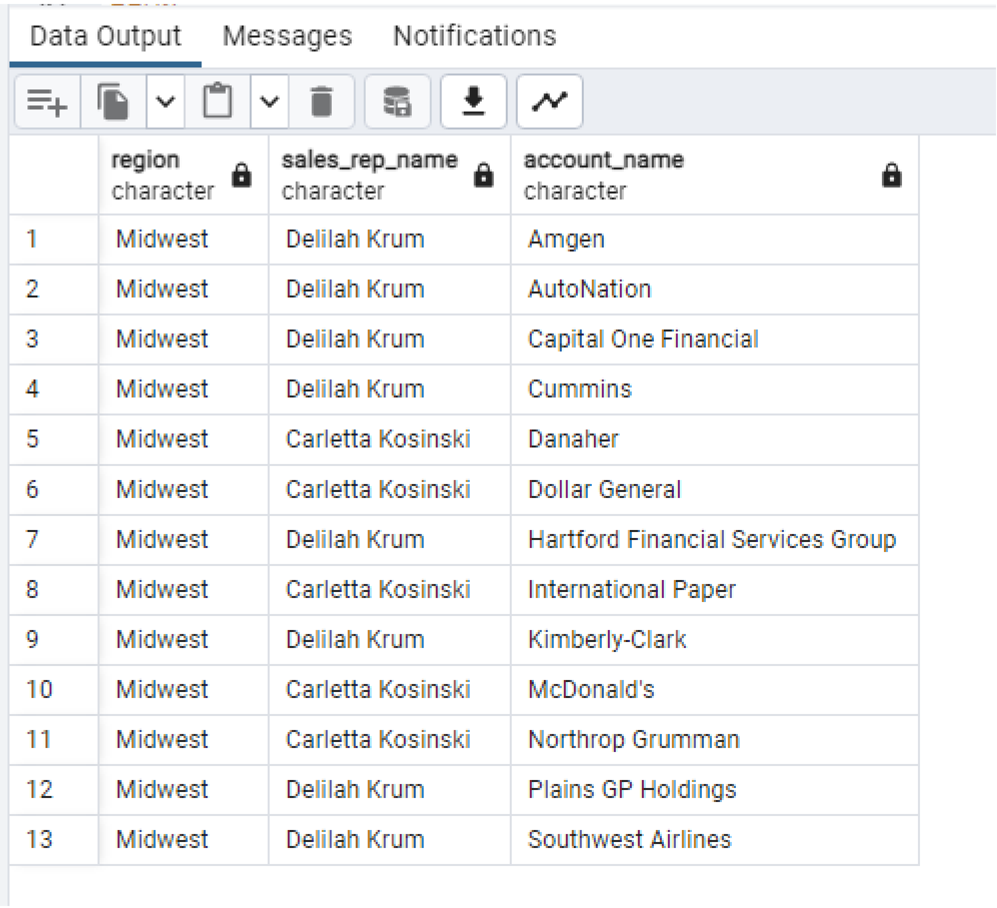
QUESTION 5

CODE:

```
SELECT r.name as region,
        s.name as sales_rep_name,
        a.name as account_name
FROM accounts a
INNER JOIN sales_reps s ON a.sales_rep_id = s.id
INNER JOIN region r ON s.region_id = r.id
WHERE (SUBSTRING(s.name, POSITION(' ' IN s.name) + 1) LIKE 'K%'
    AND
```

```
            r.name = 'Midwest')
ORDER BY account_name ASC;
```

SCREEN SHOT:



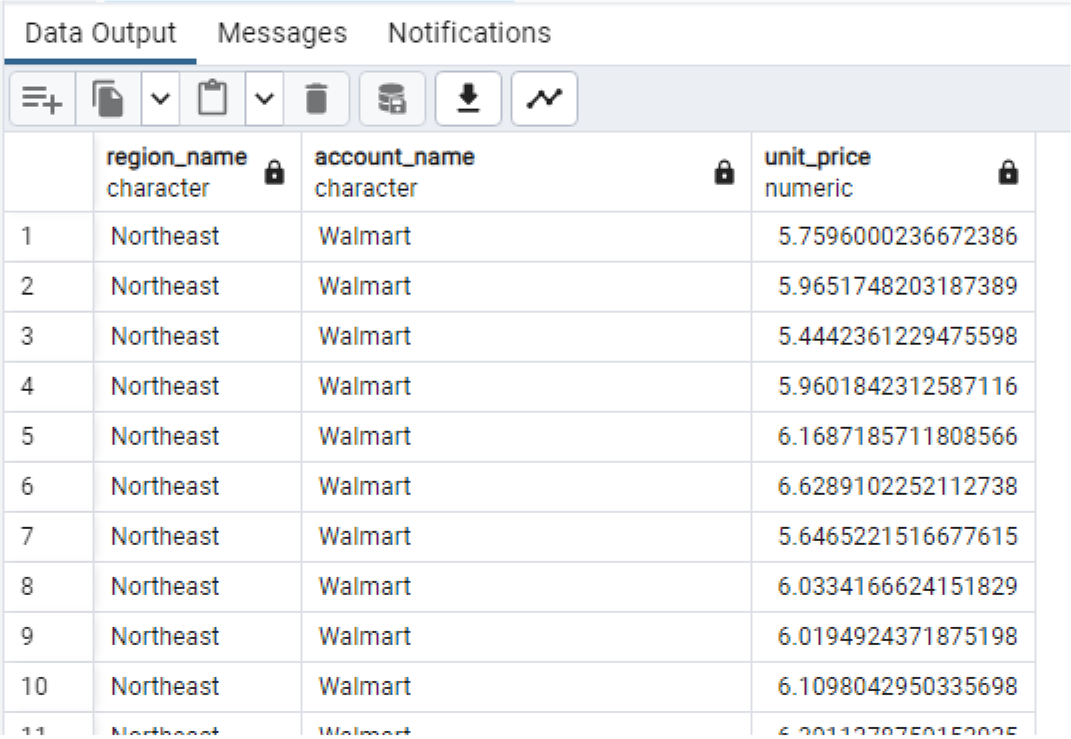
	region character	sales_rep_name character	account_name character
1	Midwest	Delilah Krum	Amgen
2	Midwest	Delilah Krum	AutoNation
3	Midwest	Delilah Krum	Capital One Financial
4	Midwest	Delilah Krum	Cummins
5	Midwest	Carletta Kosinski	Danaher
6	Midwest	Carletta Kosinski	Dollar General
7	Midwest	Delilah Krum	Hartford Financial Services Group
8	Midwest	Carletta Kosinski	International Paper
9	Midwest	Delilah Krum	Kimberly-Clark
10	Midwest	Carletta Kosinski	McDonald's
11	Midwest	Carletta Kosinski	Northrop Grumman
12	Midwest	Delilah Krum	Plains GP Holdings
13	Midwest	Delilah Krum	Southwest Airlines

QUESTION 6

CODE:

```
SELECT r.name AS region_name,
            a.name AS account_name,
            CASE
            WHEN o.total_amt_usd = 0 THEN 0
            ELSE o.total_amt_usd / (o.total + 0.01)
            END AS unit_price
FROM orders o
INNER JOIN accounts a
ON o.account_id = a.id
INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
INNER JOIN region r
ON s.region_id = r.id
WHERE o.standard_qty>100;
```

SCREEN SHOT:



	region_name character	account_name character	unit_price numeric
1	Northeast	Walmart	5.7596000236672386
2	Northeast	Walmart	5.9651748203187389
3	Northeast	Walmart	5.4442361229475598
4	Northeast	Walmart	5.9601842312587116
5	Northeast	Walmart	6.1687185711808566
6	Northeast	Walmart	6.6289102252112738
7	Northeast	Walmart	5.6465221516677615
8	Northeast	Walmart	6.0334166624151829
9	Northeast	Walmart	6.0194924371875198
10	Northeast	Walmart	6.1098042950335698
11	Northeast	Walmart	6.2011278750152025

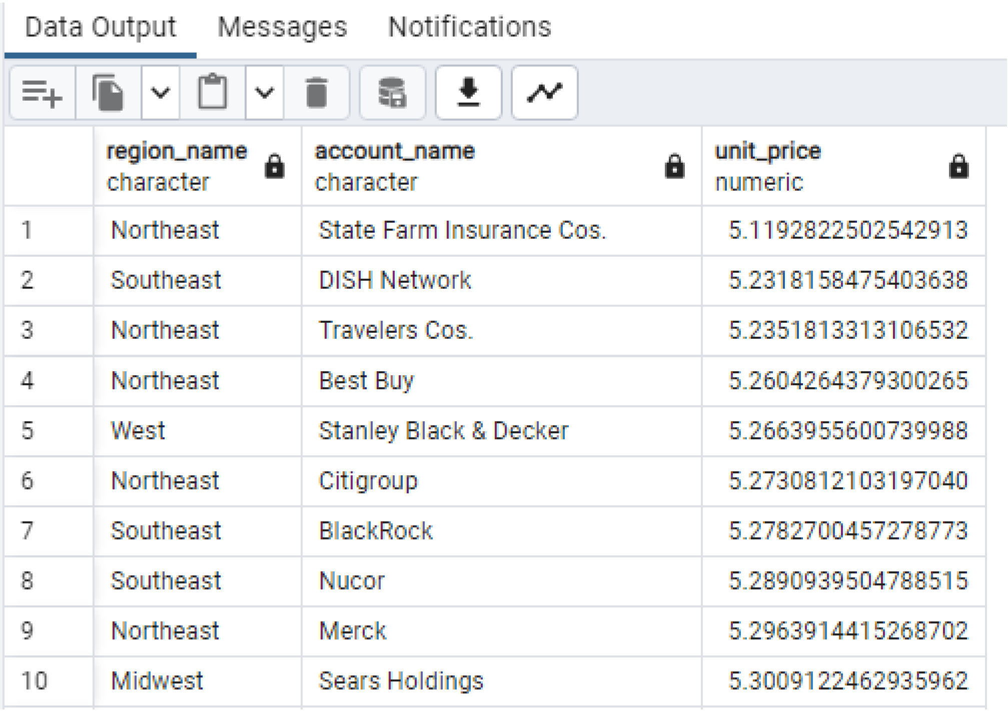
QUESTION 7

CODE:

```
SELECT r.name AS region_name,
        a.name AS account_name,
        CASE
        WHEN o.total_amt_usd = 0 THEN 0
  ELSE o.total_amt_usd / (o.total + 0.01)
        END AS unit_price
FROM orders o
INNER JOIN accounts a
ON o.account_id = a.id
INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
```

```
INNER JOIN region r

ON s.region_id = r.id

WHERE (o.standard_qty>100 AND poster_qty>50)

ORDER BY unit_price ASC;
```

SCREEN SHOT:



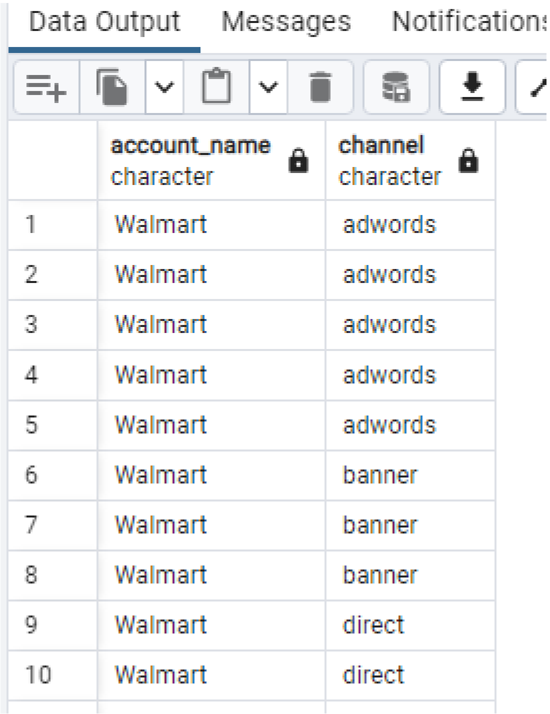
	region_name character	account_name character	unit_price numeric
1	Northeast	State Farm Insurance Cos.	5.1192822502542913
2	Southeast	DISH Network	5.2318158475403638
3	Northeast	Travelers Cos.	5.2351813313106532
4	Northeast	Best Buy	5.2604264379300265
5	West	Stanley Black & Decker	5.2663955600739988
6	Northeast	Citigroup	5.2730812103197040
7	Southeast	BlackRock	5.2782700457278773
8	Southeast	Nucor	5.2890939504788515
9	Northeast	Merck	5.2963914415268702
10	Midwest	Sears Holdings	5.3009122462935962

QUESTION 8

CODE:

```
SELECT
        a.name as account_name,
        w.channel
FROM web_events w
INNER JOIN accounts a
ON w.account_id = a.id
WHERE a.id=1001
ORDER BY w.channel;
```

SCREEN SHOT:



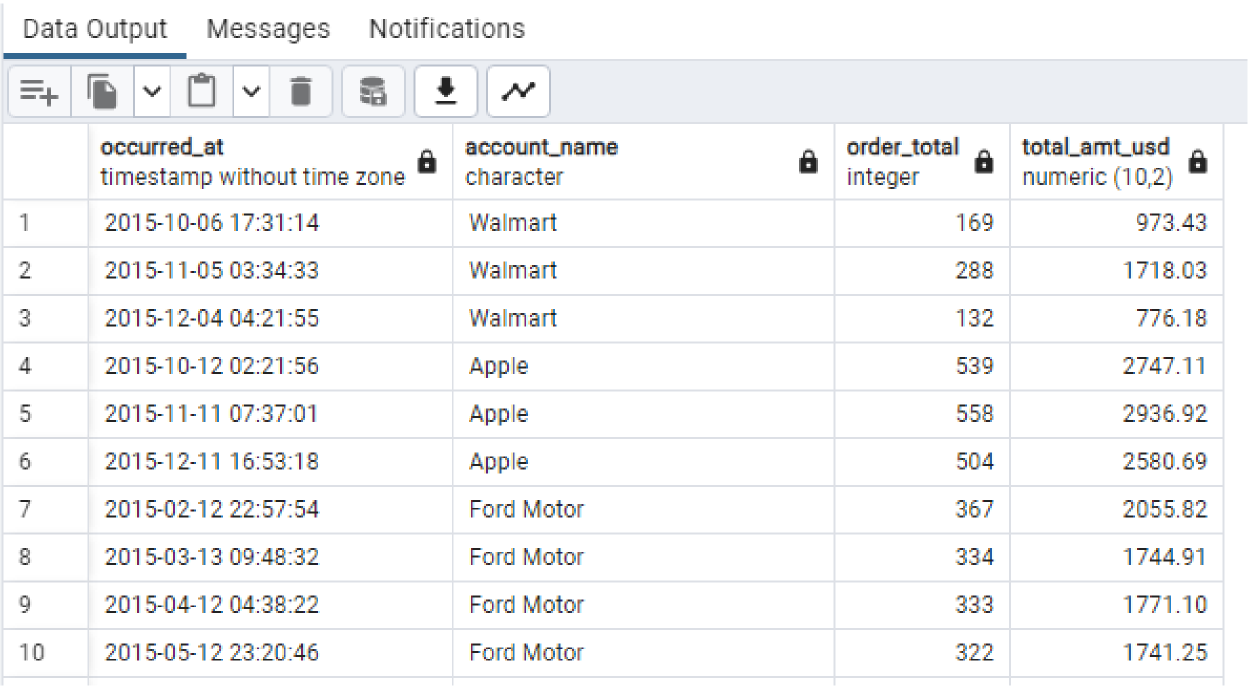
	account_name character	channel character
1	Walmart	adwords
2	Walmart	adwords
3	Walmart	adwords
4	Walmart	adwords
5	Walmart	adwords
6	Walmart	banner
7	Walmart	banner
8	Walmart	banner
9	Walmart	direct
10	Walmart	direct

QUESTION 9

CODE:

```
SELECT

        o.occurred_at,

        a.name as account_name,

        o.total as order_total,

        o.total_amt_usd

FROM orders o

INNER JOIN accounts a

ON o.account_id= a.id

WHERE EXTRACT(YEAR FROM o.occurred_at)='2015';
```

SCREEN SHOT:



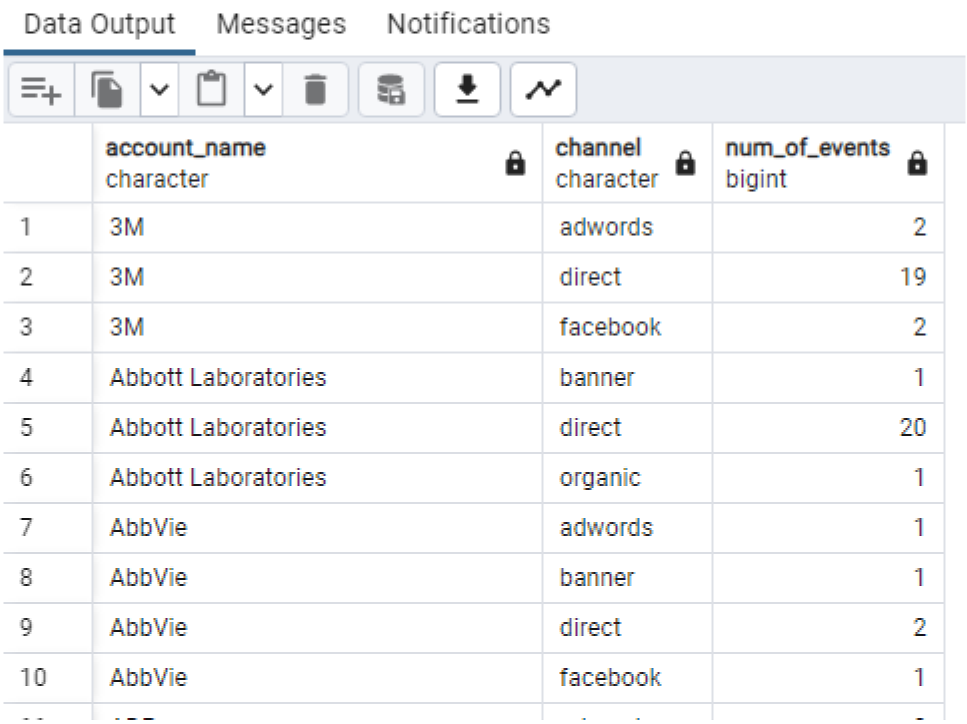
	occurred_at timestamp without time zone	account_name character	order_total integer	total_amt_usd numeric (10,2)
1	2015-10-06 17:31:14	Walmart	169	973.43
2	2015-11-05 03:34:33	Walmart	288	1718.03
3	2015-12-04 04:21:55	Walmart	132	776.18
4	2015-10-12 02:21:56	Apple	539	2747.11
5	2015-11-11 07:37:01	Apple	558	2936.92
6	2015-12-11 16:53:18	Apple	504	2580.69
7	2015-02-12 22:57:54	Ford Motor	367	2055.82
8	2015-03-13 09:48:32	Ford Motor	334	1744.91
9	2015-04-12 04:38:22	Ford Motor	333	1771.10
10	2015-05-12 23:20:46	Ford Motor	322	1741.25

QUESTION 10

CODE:

```
SELECT a.name AS account_name,
        w.channel,
        COUNT(*) AS num_of_events
FROM accounts a
INNER JOIN web_events w
ON a.id = w.account_id
GROUP BY a.name, w.channel
ORDER BY a.name, w.channel;
```

SCREEN SHOT:



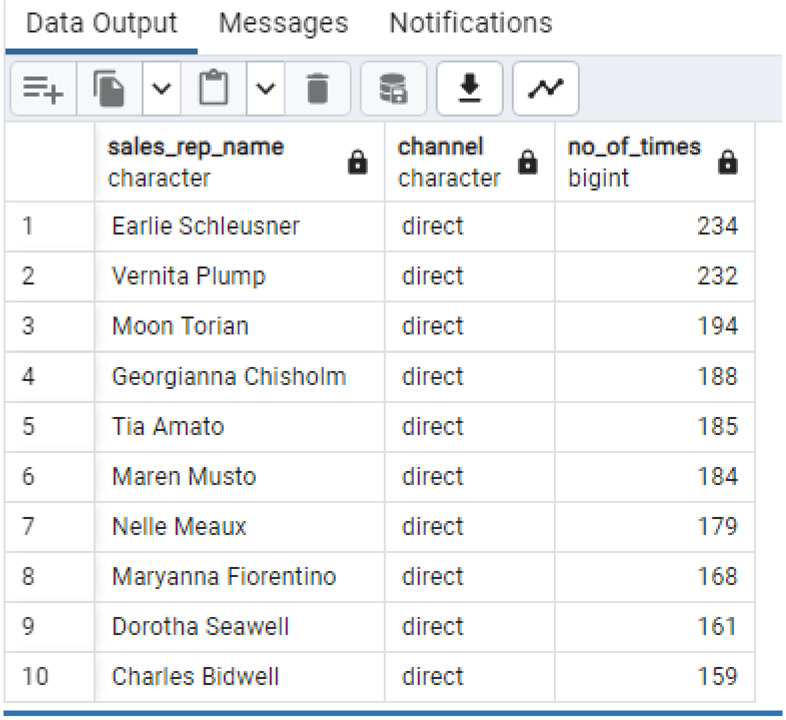
	account_name character	channel character	num_of_events bigint
1	3M	adwords	2
2	3M	direct	19
3	3M	facebook	2
4	Abbott Laboratories	banner	1
5	Abbott Laboratories	direct	20
6	Abbott Laboratories	organic	1
7	AbbVie	adwords	1
8	AbbVie	banner	1
9	AbbVie	direct	2
10	AbbVie	facebook	1

QUESTION 11

CODE:

```
SELECT
        s.name AS sales_rep_name,
        w.channel,
        COUNT(*) AS no_of_times
FROM web_events w
INNER JOIN accounts a
ON w.account_id = a.id
INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
GROUP BY s.name, w.channel
ORDER BY no_of_times DESC;
```

SCREEN SHOT:



	sales_rep_name character	channel character	no_of_times bigint
1	Earlie Schleusner	direct	234
2	Vernita Plump	direct	232
3	Moon Torian	direct	194
4	Georgianna Chisholm	direct	188
5	Tia Amato	direct	185
6	Maren Musto	direct	184
7	Nelle Meaux	direct	179
8	Maryanna Fiorentino	direct	168
9	Dorotha Seawell	direct	161
10	Charles Bidwell	direct	159

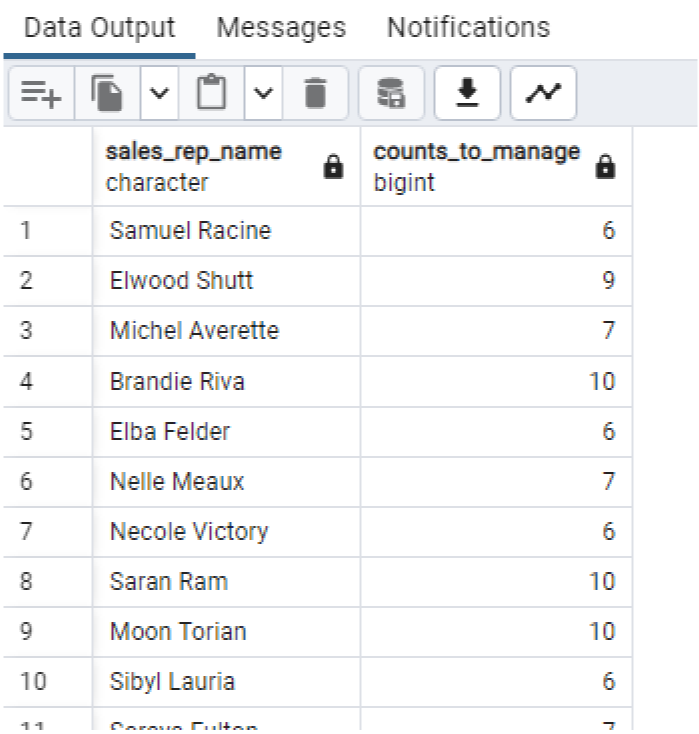
QUESTION 12

(a)

CODE:

```
SELECT
   s.name AS sales_rep_name,
   COUNT(*) AS counts_to_manage
FROM accounts a
INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
GROUP BY s.name
HAVING COUNT(*) > 5;
```

SCREEN SHOT:



	sales_rep_name character	counts_to_manage bigint
1	Samuel Racine	6
2	Elwood Shutt	9
3	Michel Averette	7
4	Brandie Riva	10
5	Elba Felder	6
6	Nelle Meaux	7
7	Necole Victory	6
8	Saran Ram	10
9	Moon Torian	10
10	Sibyl Lauria	6
11	Soraya Fulton	7

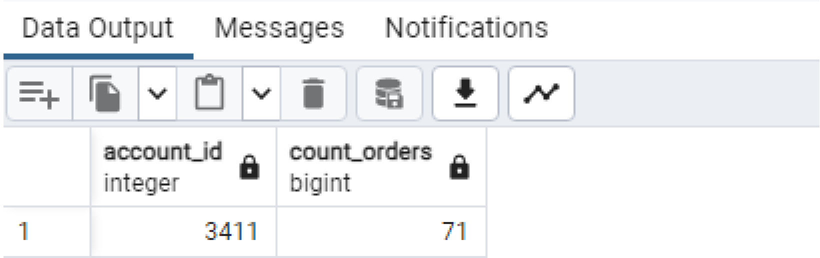
QUESTION 12

(b)

CODE:

```
SELECT
        account_id,
        COUNT(*) AS count_orders
FROM orders
GROUP BY account_id
HAVING COUNT(*)=(
        SELECT MAX(order_count)
        FROM (
                SELECT COUNT(id) as order_count
                FROM orders
                GROUP BY account_id
        ) table_
);
```

SCREEN SHOT:



	account_id integer	count_orders bigint
1	3411	71

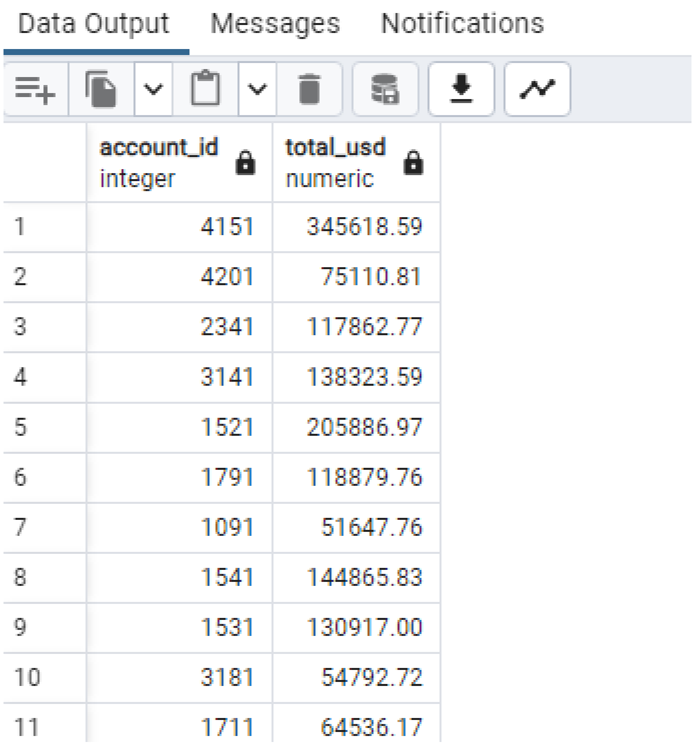
QUESTION 12

(c)

CODE:

```
SELECT
        account_id,
        SUM(total_amt_usd) as total_usd
FROM orders
GROUP BY account_id
HAVING SUM(total_amt_usd)>30000;
```

SCREEN SHOT:



	account_id integer	total_usd numeric
1	4151	345618.59
2	4201	75110.81
3	2341	117862.77
4	3141	138323.59
5	1521	205886.97
6	1791	118879.76
7	1091	51647.76
8	1541	144865.83
9	1531	130917.00
10	3181	54792.72
11	1711	64536.17

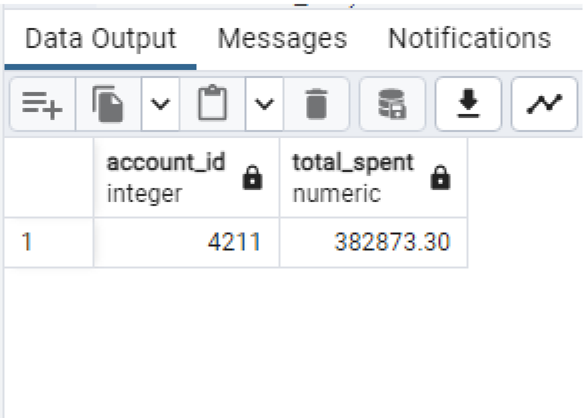
QUESTION 12

(d)

CODE:

```
SELECT
            account_id,
   SUM(total_amt_usd) AS total_spent
FROM orders
GROUP BY account_id
HAVING SUM(total_amt_usd) = (
   SELECT MAX(total_spent)
   FROM (
      SELECT SUM(total_amt_usd) AS total_spent
      FROM orders
      GROUP BY account_id
   ) table_
);
```

SCREEN SHOT:



	account_id integer	total_spent numeric
1	4211	382873.30

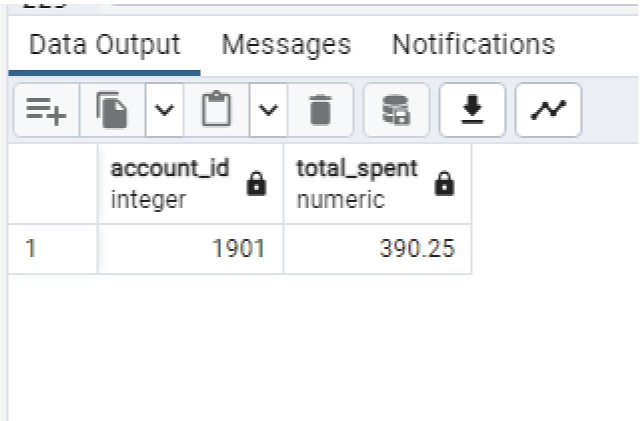
QUESTION 12

(e)

CODE:

```
SELECT
        account_id,
  SUM(total_amt_usd) AS total_spent
FROM orders
GROUP BY account_id
HAVING SUM(total_amt_usd) = (
  SELECT MIN(total_spent)
  FROM (
    SELECT SUM(total_amt_usd) AS total_spent
    FROM orders
    GROUP BY account_id
  ) table_
);
```

SCREEN SHOT:



	account_id integer	total_spent numeric
1	1901	390.25

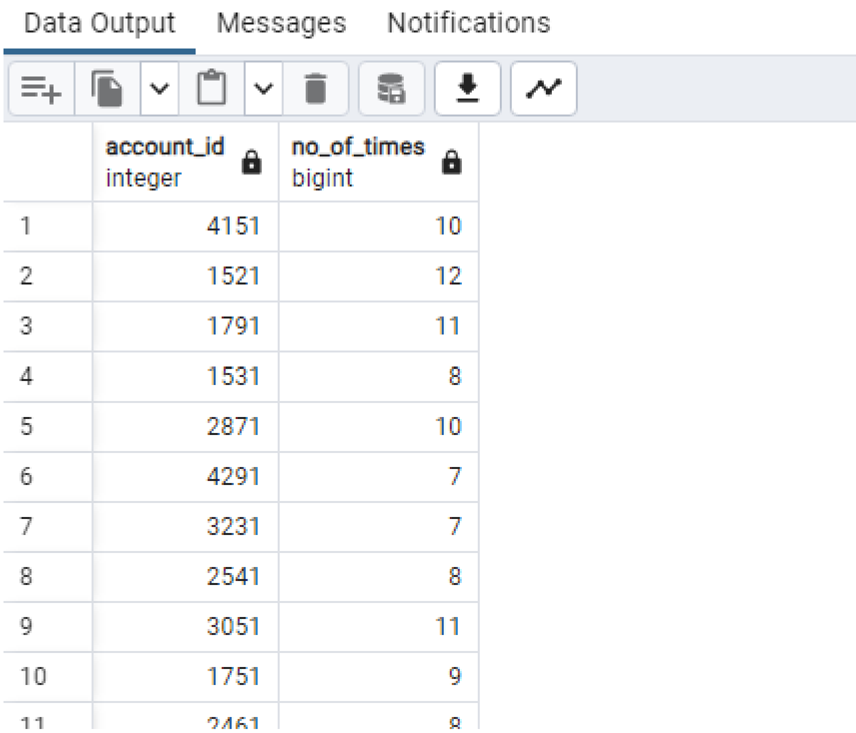
QUESTION 12

(f)

CODE:

```
SELECT
        account_id,
        COUNT(*) as no_of_times
FROM web_events
WHERE channel='facebook'
GROUP BY account_id
HAVING COUNT(*) > 6;
```

SCREEN SHOT:



	account_id integer	no_of_times bigint
1	4151	10
2	1521	12
3	1791	11
4	1531	8
5	2871	10
6	4291	7
7	3231	7
8	2541	8
9	3051	11
10	1751	9
11	2461	8

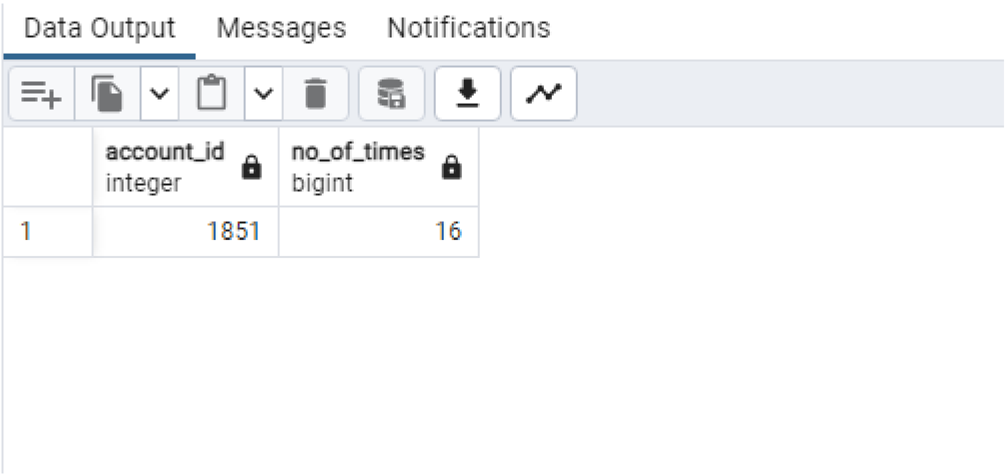
QUESTION 12

(g)

CODE:

```
SELECT
   account_id,
   COUNT(*) as no_of_times
FROM web_events
WHERE channel = 'facebook'
GROUP BY account_id
HAVING COUNT(*) = (
   SELECT MAX(count_)
   FROM (
      SELECT COUNT(*) as count_
      FROM web_events
      WHERE channel = 'facebook'
      GROUP BY account_id
   ) table_
);
```

SCREEN SHOT:



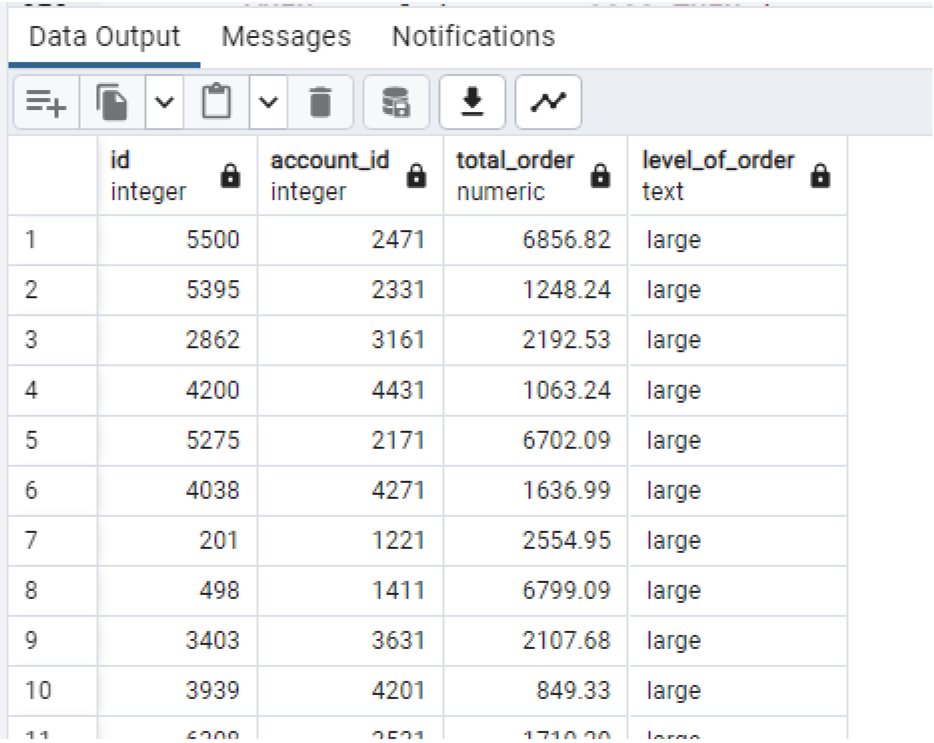
	account_id integer	no_of_times bigint
1	1851	16

QUESTION 13

CODE:

```
SELECT
        id,
        account_id,
        SUM(total_amt_usd) as total_order,
        CASE
                WHEN SUM(total_amt_usd) >= 300 THEN 'large'
                ELSE 'small'
                END AS level_of_order
FROM orders
GROUP BY id, account_id;
```

SCREEN SHOT:



	id integer	account_id integer	total_order numeric	level_of_order text
1	5500	2471	6856.82	large
2	5395	2331	1248.24	large
3	2862	3161	2192.53	large
4	4200	4431	1063.24	large
5	5275	2171	6702.09	large
6	4038	4271	1636.99	large
7	201	1221	2554.95	large
8	498	1411	6799.09	large
9	3403	3631	2107.68	large
10	3939	4201	849.33	large
11	6209	2521	1710.20	large

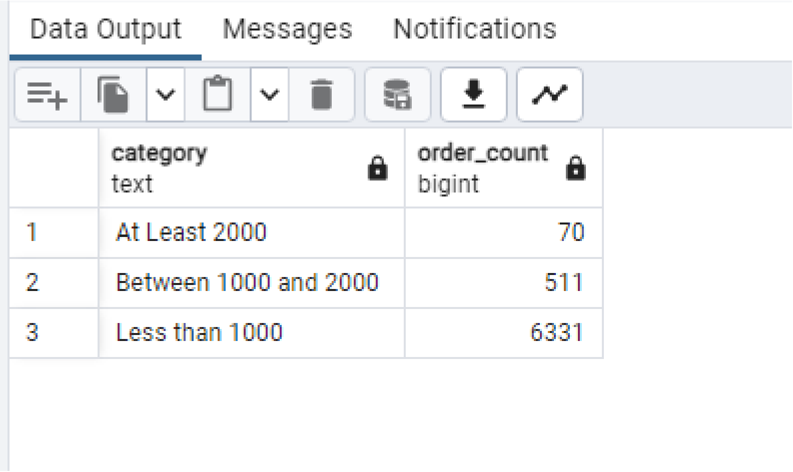
QUESTION 14

CODE

```
SELECT
   CASE
      WHEN total_items >= 2000 THEN 'At Least 2000'
      WHEN total_items BETWEEN 1000 AND 1999 THEN 'Between 1000 and 2000'
      ELSE 'Less than 1000'
   END AS category,
   COUNT(*) AS order_count
FROM (
   SELECT
```

```
    o.id AS order_id,
    SUM(o.standard_qty + o.gloss_qty + o.poster_qty) AS total_items
  FROM orders o
  GROUP BY o.id
) table_
GROUP BY category
ORDER BY category;
```

SCREEN SHOT:



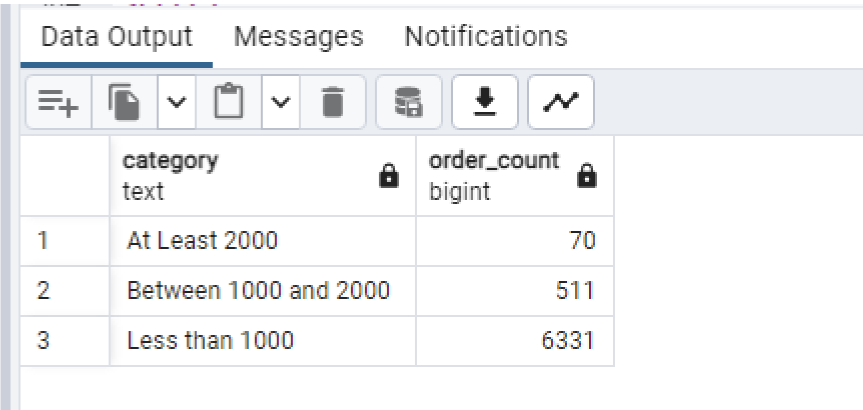
	category text	order_count bigint
1	At Least 2000	70
2	Between 1000 and 2000	511
3	Less than 1000	6331

QUESTION 15

CODE:

```
SELECT
  CASE
    WHEN total_items >= 2000 THEN 'At Least 2000'
    WHEN total_items BETWEEN 1000 AND 1999 THEN 'Between 1000 and 2000'
    ELSE 'Less than 1000'
  END AS category,
  COUNT(*) AS order_count
FROM (
    SELECT
        o.id AS order_id,
        SUM(o.standard_qty + o.gloss_qty + o.poster_qty) AS total_items
    FROM orders o
    GROUP BY o.id
) subquery
GROUP BY category
ORDER BY category;
```

SCREEN SHOT:



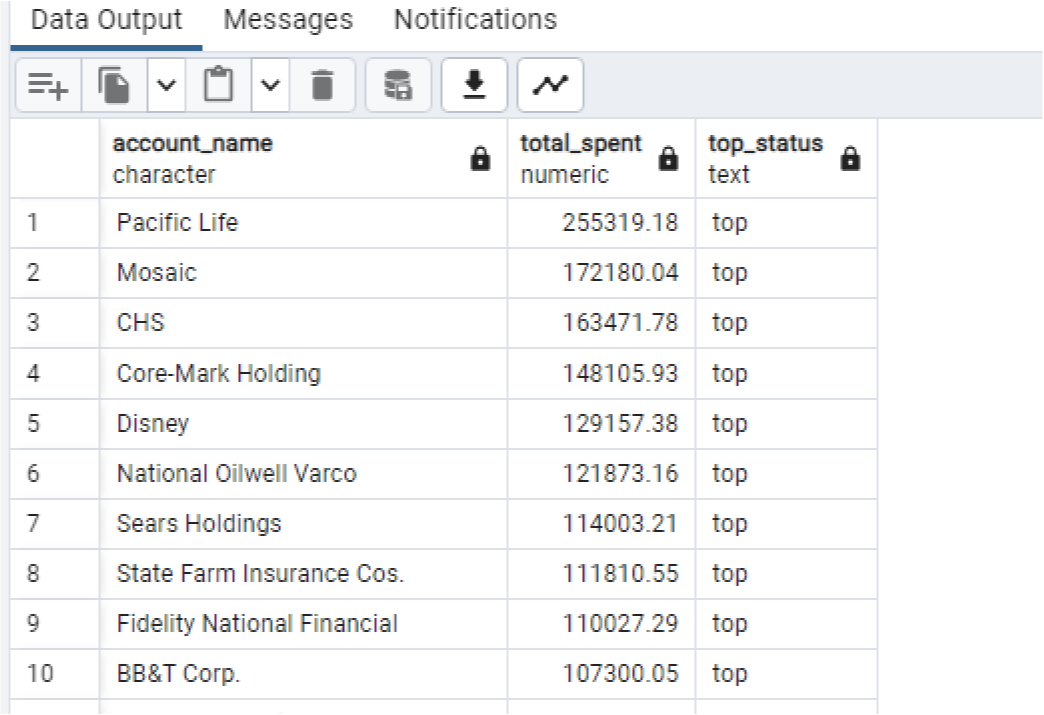
	category text	order_count bigint
1	At Least 2000	70
2	Between 1000 and 2000	511
3	Less than 1000	6331

QUESTION 16

CODE:

```
WITH CustomerSpending AS (
    SELECT
        a.id AS account_id,
        a.name AS account_name,
        SUM(CASE WHEN EXTRACT(YEAR FROM o.occurred_at) IN (2016, 2017) THEN o.total_amt_usd ELSE 0 END) AS total_spent
    FROM accounts a
    LEFT JOIN orders o ON a.id = o.account_id
    GROUP BY a.id, a.name
)
SELECT
  cs.account_name,
  SUM(cs.total_spent) AS total_spent,
  CASE
    WHEN SUM(cs.total_spent) > 20000 THEN 'top'
    ELSE 'not top'
  END AS top_status
FROM CustomerSpending cs
GROUP BY cs.account_name
HAVING SUM(cs.total_spent) > 20000
ORDER BY top_status DESC, total_spent DESC;
```

SCREEN SHOT:



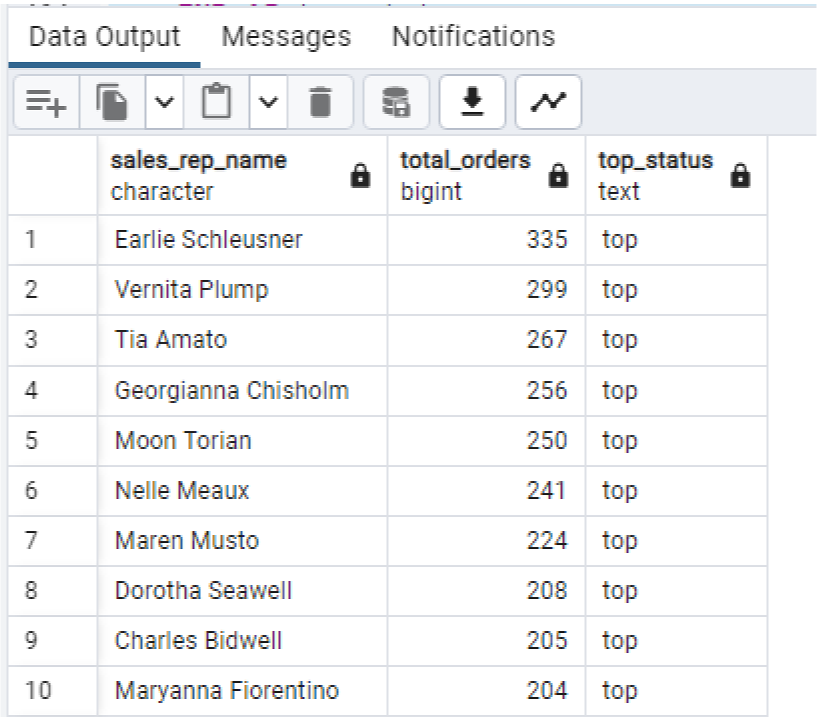
	account_name character	total_spent numeric	top_status text
1	Pacific Life	255319.18	top
2	Mosaic	172180.04	top
3	CHS	163471.78	top
4	Core-Mark Holding	148105.93	top
5	Disney	129157.38	top
6	National Oilwell Varco	121873.16	top
7	Sears Holdings	114003.21	top
8	State Farm Insurance Cos.	111810.55	top
9	Fidelity National Financial	110027.29	top
10	BB&T Corp.	107300.05	top

QUESTION 17

CODE:

```
WITH SalesRepOrders AS (
  SELECT
    s.name AS sales_rep_name,
    COUNT(o.id) AS total_orders
  FROM sales_reps s
  LEFT JOIN accounts a ON s.id = a.sales_rep_id
  LEFT JOIN orders o ON a.id = o.account_id
  GROUP BY s.name
)

SELECT
  sro.sales_rep_name,
  sro.total_orders,
  CASE
    WHEN sro.total_orders > 200 THEN 'top'
    ELSE 'not top'
  END AS top_status
FROM SalesRepOrders sro
ORDER BY top_status DESC, sro.total_orders DESC;
```

SCREEN SHOT:



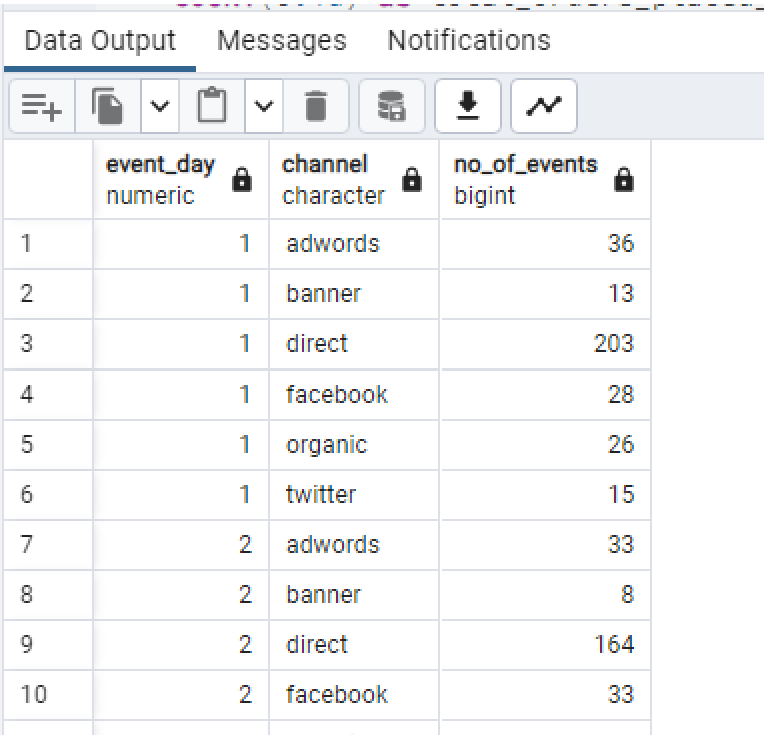
	sales_rep_name character	total_orders bigint	top_status text
1	Earlie Schleusner	335	top
2	Vernita Plump	299	top
3	Tia Amato	267	top
4	Georgianna Chisholm	256	top
5	Moon Torian	250	top
6	Nelle Meaux	241	top
7	Maren Musto	224	top
8	Dorotha Seawell	208	top
9	Charles Bidwell	205	top
10	Maryanna Fiorentino	204	top

QUESTION 18

CODE:

```
SELECT
        EXTRACT(DAY FROM occurred_at) as event_day,
        channel,
        COUNT(*) as no_of_events
FROM web_events
GROUP BY EXTRACT(DAY FROM occurred_at), channel
ORDER BY event_day;
```

SCREEN SHOT:



	event_day numeric	channel character	no_of_events bigint
1	1	adwords	36
2	1	banner	13
3	1	direct	203
4	1	facebook	28
5	1	organic	26
6	1	twitter	15
7	2	adwords	33
8	2	banner	8
9	2	direct	164
10	2	facebook	33

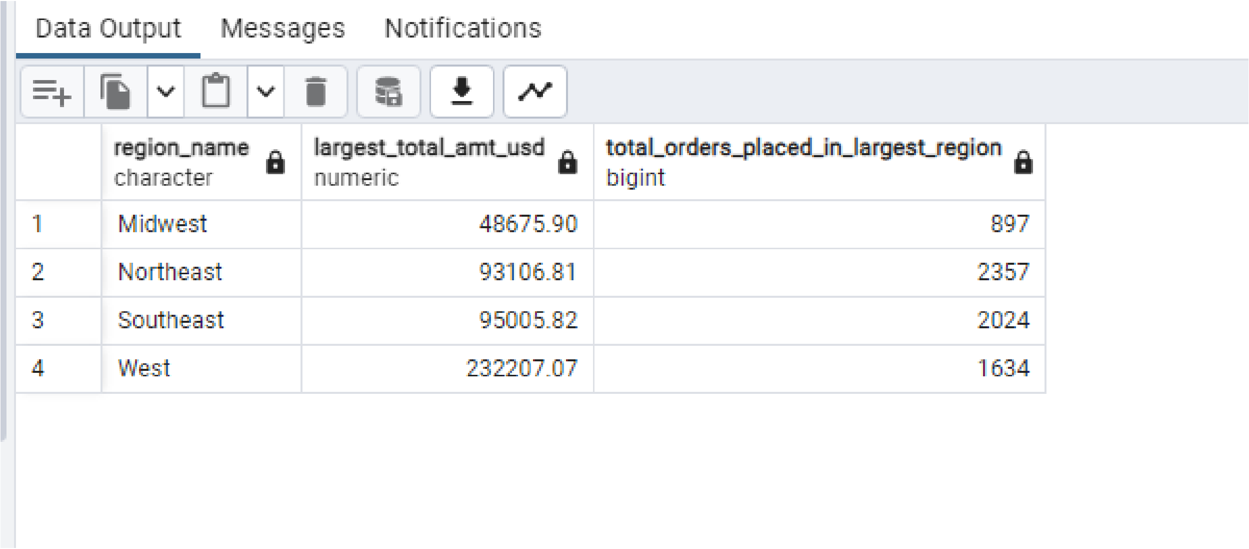
QUESTION 19

CODE:

```
SELECT
    r.name as region_name,
    MAX(o.total_amt_usd) as largest_total_amt_usd,
    COUNT(o.id) as total_orders_placed_in_largest_region
FROM orders o
INNER JOIN accounts a
ON o.account_id = a.id
INNER JOIN sales_reps s
ON a.sales_rep_id = s.id
INNER JOIN region r
```

```
ON s.region_id = r.id
GROUP BY r.name
ORDER BY r.name;
```

SCREEN SHOT:



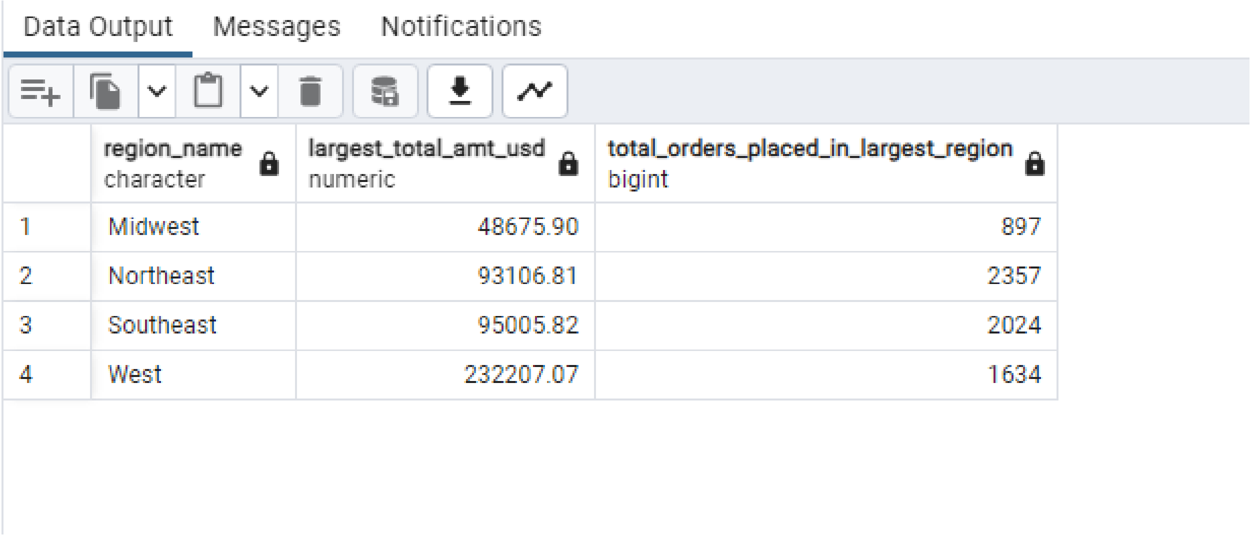
	region_name character	largest_total_amt_usd numeric	total_orders_placed_in_largest_region bigint
1	Midwest	48675.90	897
2	Northeast	93106.81	2357
3	Southeast	95005.82	2024
4	West	232207.07	1634

QUESTION 20

CODE:

```
with total_orders_placed_in_largest_region as
(
        SELECT r.name as region_name,
        MAX(o.total_amt_usd) as largest_total_amt_usd,
        COUNT(o.id) as total_orders_placed_in_largest_region
        FROM orders o
        INNER JOIN accounts a
        ON o.account_id = a.id
        INNER JOIN sales_reps s
        ON a.sales_rep_id = s.id
        INNER JOIN region r
        ON s.region_id = r.id
        GROUP BY r.name
        ORDER BY r.name
)
SELECT * FROM total_orders_placed_in_largest_region;
```

SCREEN SHOT:



	region_name character	largest_total_amt_usd numeric	total_orders_placed_in_largest_region bigint
1	Midwest	48675.90	897
2	Northeast	93106.81	2357
3	Southeast	95005.82	2024
4	West	232207.07	1634