
SNAPSHOT SUMMARY OF PARCH AND POSEY ANALYSIS PROJECT

ABOUT THE DATA

The purpose of this project was to draw insights and analyze data from the make-believe paper store called Parch & Posey. The data for this project came from Udacity's Data Science nanodegree program.

video.udacity-data.com/topher/2020/May/5eb5533b_parch-and-posey/parch-and-posey.sql

I formulated basic and complex queries using advanced functions like window functions, subqueries, and common table expressions to answer insightful questions for the company. I also created a dynamic dashboard for its shareholders as well.

```
BEGIN TRANSACTION;
CREATE TABLE web_events (
  id integer,
  account_id integer,
  occurred_at timestamp,
  channel bpchar
);
INSERT INTO web_events VALUES (1,1001,'2015-10-06 17:13:58','direct');
INSERT INTO web_events VALUES (2,1001,'2015-11-05 03:08:26','direct');
INSERT INTO web_events VALUES (3,1001,'2015-12-04 03:57:24','direct');
INSERT INTO web_events VALUES (4,1001,'2016-01-02 00:55:03','direct');
INSERT INTO web_events VALUES (5,1001,'2016-02-01 19:02:33','direct');
INSERT INTO web_events VALUES (6,1001,'2016-03-02 15:15:22','direct');
INSERT INTO web_events VALUES (7,1001,'2016-04-01 10:58:55','direct');
INSERT INTO web_events VALUES (8,1001,'2016-05-01 15:26:44','direct');
INSERT INTO web_events VALUES (9,1001,'2016-05-31 20:53:47','direct');
INSERT INTO web_events VALUES (10,1001,'2016-06-30 12:09:45','direct');
INSERT INTO web_events VALUES (11,1001,'2016-07-30 03:06:26','direct');
INSERT INTO web_events VALUES (12,1001,'2016-08-28 06:42:42','direct');
INSERT INTO web_events VALUES (13,1001,'2016-09-26 23:14:59','direct');
INSERT INTO web_events VALUES (14,1001,'2016-10-26 20:21:09','direct');
INSERT INTO web_events VALUES (15,1001,'2016-11-25 22:52:29','direct');
INSERT INTO web_events VALUES (16,1001,'2016-12-24 05:35:14','direct');
INSERT INTO web_events VALUES (17,1011,'2016-12-21 10:29:36','direct');
INSERT INTO web_events VALUES (18,1021,'2015-10-12 02:10:54','direct');
INSERT INTO web_events VALUES (19,1021,'2015-11-11 07:10:19','direct');
INSERT INTO web_events VALUES (20,1021,'2015-12-11 16:28:27','direct');
INSERT INTO web_events VALUES (21,1021,'2016-01-10 09:04:12','direct');
INSERT INTO web_events VALUES (22,1021,'2016-02-09 00:46:47','direct');
```

DASHBOARD PRESENTATION



PARCH AND POSEY SALES DASHBOARD

Year

All

Region

All

\$23.14M

Total Sales

3.68M

Papers Sold

6912

Accounts

50

Sales Reps

724K

Poster Qty

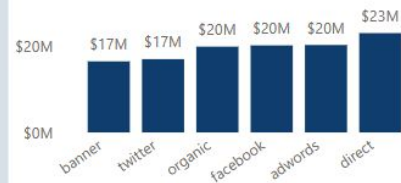
1M

Gloss Qty

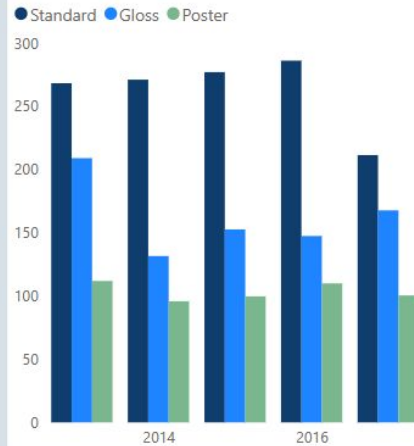
2M

Standard Qty

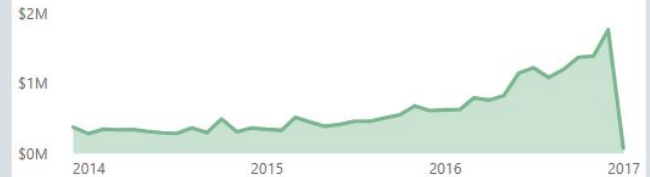
Sales Performance by Channel



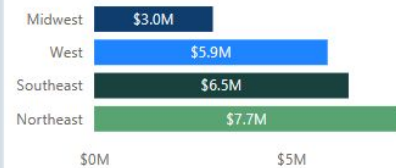
Average Paper Qty Sold per Year



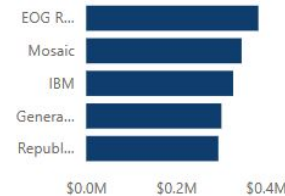
Yearly Sales Performance



Sales Performance by Region



Top Performing Accounts



Top Performing Sales Rep



EXAMPLE QUERIES

Query 1: For the region with the largest (sum) of sales total_amt_usd, how many total (count) orders were placed?

Query 2: What is the top channel used by each account to market products? How often was the same channel used?

Query 3: Have any sales reps worked on more than one account?

QUERY 1

```
SELECT t3.region,
       t2.max_sales,
       t3.tot_orders
FROM
  (SELECT r.name region,
         SUM(total_amt_usd) tot_sales,
         COUNT(*) tot_orders
   FROM region r
   JOIN sales_reps sr ON r.id = sr.region_id
   JOIN accounts a ON a.sales_rep_id = sr.id
   JOIN orders o ON o.account_id = a.id
   GROUP BY 1) T3
JOIN
  (SELECT t1.region,
         t1.tot_orders,
         MAX(tot_sales) max_sales
   FROM
     (SELECT r.name region,
            SUM(total_amt_usd) tot_sales,
            COUNT(*) tot_orders
     FROM region r
     JOIN sales_reps sr ON r.id = sr.region_id
     JOIN accounts a ON a.sales_rep_id = sr.id
     JOIN orders o ON o.account_id = a.id
     GROUP BY 1) t1
   GROUP BY 1,
          2) t2 ON t2.tot_orders = t3.tot_orders
ORDER BY 2 DESC,
        3
```

QUERY 2

```
SELECT t3.id,
       t3.name,
       t3.channel,
       t3.ct
FROM
  (SELECT a.id,
         a.name,
         channel,
         COUNT(*) ct
   FROM accounts a
   JOIN web_events we ON a.id = we.account_id
   GROUP BY 1,
          2,
          3) t3
JOIN
  (SELECT t1.id,
         t1.name,
         t1.channel,
         MAX(ct) MAX
   FROM
     (SELECT a.id,
            a.name,
            channel,
            COUNT(*) ct
     FROM accounts a
     JOIN web_events we ON we.account_id = a.id
     GROUP BY 1,
            2,
            3) t1
   GROUP BY 1,
          2,
          3) t2 ON t2.id = t3.id
AND t3.ct = t2.max
ORDER BY 2,
        4
```

QUERY 3

```
SELECT sr.name rep,
       COUNT(a.*) accounts_worked
FROM accounts a
JOIN sales_reps sr ON sr.id = a.sales_rep_id
GROUP BY rep
```