

WITH

-- create a table for every customer for all months using cross join

```
customer_alldate AS
(
    SELECT
        CUST_CODE,
        year_month
    FROM (
        SELECT FORMAT_DATETIME("%Y-%m", master_date) AS year_month
        FROM UNNEST(
            GENERATE_DATE_ARRAY(
                (SELECT MIN(SHOP_DATE) FROM
                (SELECT PARSE_DATE('%Y%m%d', CAST(SHOP_DATE AS STRING)) AS
SHOP_DATE,
                    FROM `axiomatic-grove-331409.Supermarket2.supermarket`
                    WHERE CUST_CODE IS NOT NULL)),
                (SELECT DATE_ADD(MAX(SHOP_DATE),INTERVAL 1 MONTH) FROM
                (SELECT PARSE_DATE('%Y%m%d', CAST(SHOP_DATE AS STRING)) AS
SHOP_DATE,
                    FROM `axiomatic-grove-331409.Supermarket2.supermarket`
                    WHERE CUST_CODE IS NOT NULL)), INTERVAL 1 MONTH)) AS
master_date )
    CROSS JOIN
        (SELECT DISTINCT(CUST_CODE),
        FROM `axiomatic-grove-331409.Supermarket2.supermarket`
        WHERE CUST_CODE IS NOT NULL)
    ) ,
```

-- create customer table where txn occurred groupby into monthly level

```
customer_table AS
(
    SELECT
        CUST_CODE,
        FORMAT_DATETIME("%Y-%m", (PARSE_DATE('%Y%m%d', CAST(SHOP_DATE AS
STRING)))) AS year_month,
        FROM `axiomatic-grove-331409.Supermarket2.supermarket`
        WHERE CUST_CODE IS NOT NULL
        GROUP BY CUST_CODE, year_month
    ) ,
```

-- combine the first two tables to fill in missing months for each customer

```
combined_table AS
(
    SELECT
        customer_alldate.CUST_CODE AS CUST_CODE1,
        customer_table.CUST_CODE AS CUST_CODE2,
        customer_alldate.year_month AS year_month1,
```

```

        customer_table.year_month AS year_month2,
    FROM customer_alldate
    LEFT JOIN customer_table
    ON customer_alldate.CUST_CODE = customer_table.CUST_CODE AND
customer_alldate.year_month = customer_table.year_month
    ORDER BY CUST_CODE1,year_month1
) ,

```

-- create last visited month lag, calculate days from last visit month, and flag first purchased month

```

lag_combined_table AS
(
    SELECT *,
        LAG(year_month2) OVER (PARTITION BY CUST_CODE1 ORDER BY year_month2) AS
prev_month,
        DATE_DIFF(DATETIME(CONCAT(year_month2,"-01")),
DATETIME(CONCAT((LAG(year_month2) OVER (PARTITION BY CUST_CODE1 ORDER BY
year_month2)),"-01")), DAY) AS days,
        MIN(year_month2) OVER (PARTITION BY CUST_CODE1 ORDER BY year_month1
                                ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW)
AS filled_min_date
    FROM combined_table
    ORDER BY CUST_CODE1,year_month1
)

```

-- Flag status and numerical value and removing months prior to first purchase

```

SELECT * ,
    CASE
        WHEN prev_month IS NULL AND year_month2 IS NOT NULL THEN "New"
        WHEN days IS NULL THEN "Churn"
        WHEN days <= 31 THEN "Active"
        WHEN days > 31 THEN "Reactivate"
        ELSE "Churn"
    END AS status,
    CASE
        WHEN prev_month IS NULL AND year_month2 IS NOT NULL THEN 1
        WHEN days IS NULL THEN -1
        WHEN days <= 31 THEN 1
        WHEN days > 31 THEN 1
        ELSE -1
    END AS status_flag
FROM lag_combined_table
WHERE filled_min_date IS NOT NULL
ORDER BY CUST_CODE1,year_month1

```

Query complete (28.9 sec elapsed, 517.7 MB processed)

Job information [Results](#) JSON Execution details

48	CUST0000000689	CUST0000000689	2007-07	2007-07	null	null	2007-07	New	1
49	CUST0000000689	CUST0000000689	2007-08	2007-08	2007-07	31	2007-07	Active	1
50	CUST0000000689	null	2007-09	null	null	null	2007-07	Churn	-1
51	CUST0000000689	CUST0000000689	2007-10	2007-10	2007-08	61	2007-07	Reactivate	1
52	CUST0000000689	CUST0000000689	2007-11	2007-11	2007-10	31	2007-07	Active	1
53	CUST0000000689	CUST0000000689	2007-12	2007-12	2007-11	30	2007-07	Active	1
54	CUST0000000689	CUST0000000689	2008-01	2008-01	2007-12	31	2007-07	Active	1
55	CUST0000000689	null	2008-02	null	null	null	2007-07	Churn	-1
56	CUST0000000689	null	2008-03	null	null	null	2007-07	Churn	-1
57	CUST0000000689	CUST0000000689	2008-04	2008-04	2008-01	91	2007-07	Reactivate	1
58	CUST0000000689	CUST0000000689	2008-05	2008-05	2008-04	30	2007-07	Active	1
59	CUST0000000689	CUST0000000689	2008-06	2008-06	2008-05	31	2007-07	Active	1
60	CUST0000000689	null	2008-07	null	null	null	2007-07	Churn	-1
61	CUST0000000998	CUST0000000998	2006-05	2006-05	null	null	2006-05	New	1
62	CUST0000000998	null	2006-06	null	null	null	2006-05	Churn	-1
63	CUST0000000998	CUST0000000998	2006-07	2006-07	2006-05	61	2006-05	Reactivate	1
64	CUST0000000998	null	2006-08	null	null	null	2006-05	Churn	-1
65	CUST0000000998	null	2006-09	null	null	null	2006-05	Churn	-1
66	CUST0000000998	null	2006-10	null	null	null	2006-05	Churn	-1
67	CUST0000000998	null	2006-11	null	null	null	2006-05	Churn	-1

Rows per page:

100 ▾

1 - 100 of 123035

First page



