Queries –

- q.1) What are the top 5 brands by receipts scanned for most recent month?

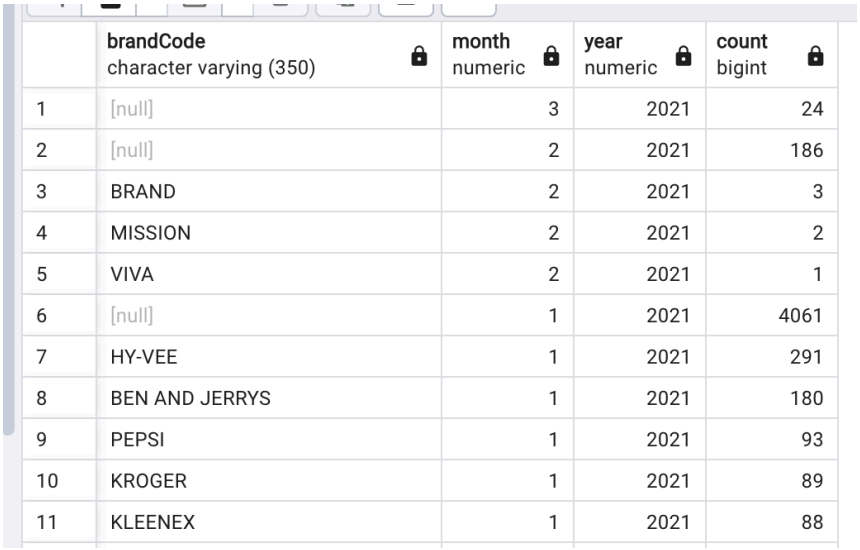
```
WITH cte1 AS (
   SELECT a.*, b."dateScanned"
   FROM receipts_line_item a
   INNER JOIN receipts_fact b ON a.receipt_id = b.receipt_id
   WHERE (EXTRACT(year FROM b."dateScanned"), EXTRACT(month FROM b."dateScanned")) =
(
      SELECT EXTRACT(year FROM MAX("dateScanned")), EXTRACT(month FROM
MAX("dateScanned"))
      FROM receipts_fact
   )
),
cte AS (
   SELECT "brandCode",
         EXTRACT(month FROM x."dateScanned") AS month,
         EXTRACT(year FROM x."dateScanned") AS year,
         COUNT(x.receipt_id) AS receipt_count
   FROM cte1 x
   GROUP BY "brandCode", EXTRACT(month FROM x."dateScanned"), EXTRACT(year FROM
x."dateScanned")
   ORDER BY receipt_count DESC
)
SELECT *
FROM cte
LIMIT 5
```

Data Output | Messages | Notifications

	brandCode character varying (350)	month numeric	year numeric	receipt_count bigint
1	[null]	3	2021	24

Answer is above query. Below query just to show that most recent month as NULL brandcodes, but for previous months we have some data –

```
with cte1 as(
	select a.*, b."dateScanned"
	from receipts_line_item a
	inner join receipts_fact b
	on a.receipt_id = b.receipt_id
--	where a."brandCode" is NOT NULL
),
cte as(
	select "brandCode", extract(month from x."dateScanned") as month,extract(year from
x."dateScanned") as year, count(x.receipt_id)
	from cte1 x
	group by "brandCode", extract(month from x."dateScanned"), extract(year from
x."dateScanned")
	order by year desc, month desc, count desc
)
select *
from cte
```

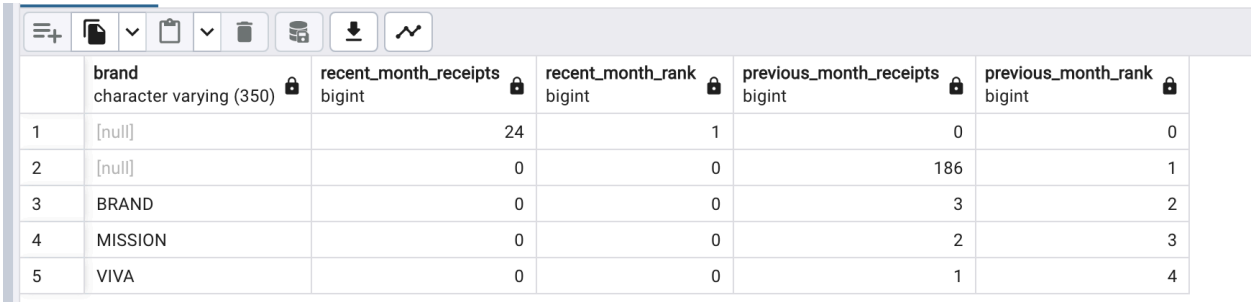


	brandCode character varying (350)	month numeric	year numeric	count bigint
1	[null]	3	2021	24
2	[null]	2	2021	186
3	BRAND	2	2021	3
4	MISSION	2	2021	2
5	VIVA	2	2021	1
6	[null]	1	2021	4061
7	HY-VEE	1	2021	291
8	BEN AND JERRYS	1	2021	180
9	PEPSI	1	2021	93
10	KROGER	1	2021	89
11	KLEENEX	1	2021	88

- q.2) How does the ranking of the top 5 brands by receipts scanned for the recent month compare to the ranking for the previous month?

```
WITH recent_month AS (
   SELECT "brandCode",
        COUNT(x.receipt_id) AS recent_month_receipts,
        ROW_NUMBER() OVER (ORDER BY COUNT(x.receipt_id) DESC) AS recent_month_rank
   FROM receipts_line_item x
   INNER JOIN receipts_fact y ON x.receipt_id = y.receipt_id
   WHERE (EXTRACT(year FROM y."dateScanned"), EXTRACT(month FROM y."dateScanned")) = (
      SELECT EXTRACT(year FROM MAX("dateScanned")), EXTRACT(month FROM
MAX("dateScanned"))
      FROM receipts_fact
   )
   GROUP BY "brandCode"
),
previous_month AS (
   SELECT "brandCode",
        COUNT(x.receipt_id) AS previous_month_receipts,
        ROW_NUMBER() OVER (ORDER BY COUNT(x.receipt_id) DESC) AS previous_month_rank
   FROM receipts_line_item x
   INNER JOIN receipts_fact y ON x.receipt_id = y.receipt_id
   WHERE (EXTRACT(year FROM y."dateScanned"), EXTRACT(month FROM y."dateScanned")) = (
      SELECT EXTRACT(year FROM MAX("dateScanned") - INTERVAL '1 month'),
           EXTRACT(month FROM MAX("dateScanned") - INTERVAL '1 month')
      FROM receipts_fact
   )
   GROUP BY "brandCode"
)
SELECT "brandCode" AS brand,
     recent_month_receipts AS recent_month_receipts,
     recent_month_rank AS recent_month_rank,
     0 AS previous_month_receipts,
     0 AS previous_month_rank
FROM recent_month
where recent_month_rank <= 5
UNION ALL
SELECT "brandCode" AS brand,
     0 AS recent_month_receipts,
     0 AS recent_month_rank,
     previous_month_receipts AS previous_month_receipts,
     previous_month_rank AS previous_month_rank
FROM previous_month
```

```
where previous_month_rank <= 5
ORDER BY recent_month_rank DESC;
```



	brand character varying (350)	recent_month_receipts bigint	recent_month_rank bigint	previous_month_receipts bigint	previous_month_rank bigint
1	[null]	24	1	0	0
2	[null]	0	0	186	1
3	BRAND	0	0	3	2
4	MISSION	0	0	2	3
5	VIVA	0	0	1	4

We get this result since, most recent month has just NULL brandcode. My assumption of this question is, that we need to compare the top 5 brands rank for recent and recent-1 month. So basically, finding top 5 brands for both months and UNION.

The other interpretation could have been to find the top 5 brands for recent month and its corresponding rank in previous month, but I have gone with the above stated interpretation.

- q.3) When considering *average spend* from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?

Accepted is greater

```
WITH accepted_receipts AS (
  SELECT
    AVG("totalSpent") AS avg_accepted_spend
  FROM
    receipts_fact
  WHERE
    "rewardsReceiptStatus" = 'FINISHED'
),
rejected_receipts AS (
  SELECT
    AVG("totalSpent") AS avg_rejected_spend
  FROM
    receipts_fact
  WHERE
    "rewardsReceiptStatus" = 'REJECTED'
)
SELECT
  'Accepted' AS receipt_status,
  avg_accepted_spend AS avg_spend
FROM
  accepted_receipts
UNION ALL
SELECT
  'Rejected' AS receipt_status,
  avg_rejected_spend AS avg_spend
FROM
  rejected_receipts
ORDER BY
  avg_spend DESC;
```

	receipt_status text	avg_spend double precision
1	Accepted	80.85430501930502
2	Rejected	23.326056338028184

- q.4) When considering *total number of items purchased* from receipts with 'rewardsReceiptStatus' of 'Accepted' or 'Rejected', which is greater?

Accepted is greater

```
WITH accepted_receipts AS (
  SELECT
    SUM("purchasedItemCount") AS total_accepted_items
  FROM
    receipts_fact
  WHERE
    "rewardsReceiptStatus" = 'FINISHED'
),
rejected_receipts AS (
  SELECT
    SUM("purchasedItemCount") AS total_rejected_items
  FROM
    receipts_fact
  WHERE
    "rewardsReceiptStatus" = 'REJECTED'
)
SELECT
  'Accepted' AS receipt_status,
  total_accepted_items as total_items
FROM
  accepted_receipts
UNION ALL
SELECT
  'Rejected' AS receipt_status,
  total_rejected_items as total_items
FROM
  rejected_receipts;
```

	receipt_status text	total_items double precision
1	Accepted	8184
2	Rejected	173

- q.5) Which brand has the most *spend* among users who were created within the past 6 months?

```
WITH recent_users AS (
        SELECT user_id
        FROM users
        WHERE "createdDate" >= (
                SELECT MAX("createdDate") - INTERVAL '6 months'
                FROM users)
        ),
spend_by_brand AS (
  SELECT
    rl."brandCode" AS brand_code,
    SUM(rf."totalSpent") AS total_spend
  FROM
    receipts_fact rf
  INNER JOIN
    recent_users ru ON rf.user_id = ru.user_id
  INNER JOIN
    receipts_line_item rl ON rf.receipt_id = rl.receipt_id
  GROUP BY
    rl."brandCode"
)
SELECT
  sb.brand_code,
  sb.total_spend
FROM
  spend_by_brand sb
ORDER BY
  sb.total_spend DESC
LIMIT 2
```

	brand_code character varying (350)	total_spend double precision
1	[null]	2930282.4200000465
2	BEN AND JERRYS	197337.68000000025

- q.6) Which brand has the most *transactions* among users who were created within the past 6 months?

```
WITH recent_users AS (
        SELECT user_id
        FROM users
        WHERE "createdDate" >= (
                SELECT MAX("createdDate") - INTERVAL '6 months'
                FROM users)
),
transactions AS (
  SELECT
    rl."brandCode",
    COUNT(rf.receipt_id) AS transaction_count
  FROM
    receipts_fact rf
  INNER JOIN
    receipts_line_item rl ON rl.receipt_id = rf.receipt_id
      INNER JOIN
    recent_users ru ON rf.user_id = ru.user_id
  GROUP BY
    rl."brandCode"
)
SELECT
  t."brandCode",
  t.transaction_count
FROM
  transactions t
ORDER BY
  t.transaction_count DESC
```

	brandCode character varying (350)	transaction_count bigint
1	[null]	2412
2	HY-VEE	291
3	BEN AND JERRYS	100
4	PEPSI	74
5	KLEENEX	70
6	KNORR	60