```
# Objective: retrieve, sort & aggregate enterprise data from the
Dillard's database (>120M transactions) #
# Tool: Teradata #


*************************************************************************
EXERCISE 2 - How many distinct skus have the brand “Polo fas”, and are
either size “XXL” or “black” in color?
*************************************************************************

SELECT COUNT(DISTINCT sku)
FROM skuinfo
WHERE brand='POLO fas' AND (size='XXL' OR color='black')


*************************************************************************
EXERCISE 3 - There was one store in the database which had only 11
days in one of its months (in other words, that store/month/year
combination only contained 11 days of transaction data). In what city
and state was this store located?
*************************************************************************
SELECT
  EXTRACT(YEAR from trnsact.saledate) as year_num,
  EXTRACT(MONTH from trnsact.saledate) as month_num,
  COUNT(DISTINCT EXTRACT(DAY from trnsact.saledate)) as date_num,
  trnsact.store,
  store_msa.city,
  store_msa.state
FROM trnsact JOIN store_msa
 ON trnsact.store=store_msa.store
GROUP BY month_num, year_num, trnsact.store, store_msa.city,
store_msa.state
WHERE trnsact.stype='P'
ORDER BY date_num ASC


*************************************************************************
EXERCISE 4 - Which sku number had the greatest increase in total sales
revenue from November to December?
*************************************************************************

SELECT
  (seasonal.nextmonth_sales-seasonal.month_sales) AS seasonal_trend,
  Seasonal.sku
FROM (
SELECT
   SUM(CASE WHEN aggregator.month_num=11 THEN aggregator.total_revenue
END) AS month_sales
   , SUM(CASE WHEN aggregator.month_num=12 THEN
```

```
aggregator.total_revenue END) AS nextmonth_sales
   , aggregator.sku
FROM
(SELECT
 SUM(amt) AS total_revenue
 ,sku
 ,EXTRACT(YEAR from saledate) as year_num
 ,EXTRACT(MONTH from saledate) as month_num
 ,TRIM(EXTRACT(YEAR from trnsact.saledate))||TRIM(EXTRACT(MONTH from
trnsact.saledate)) AS yearmonth
 ,COUNT(DISTINCT saledate) as date_num
 FROM trnsact
GROUP BY year_num, month_num, sku
WHERE stype='P' AND yearmonth<>'20058'
HAVING date_num>=20) AS aggregator
GROUP BY aggregator.sku) AS seasonal
ORDER BY seasonal_trend DESC;

**************************************************************************
*
EXERCISE 5 - What vendor has the greatest number of distinct skus in
the transaction table that do not exist in the skstinfo table?
**************************************************************************
*
## STEP 1 - joining info from skuinfo (vendors list) with trnsact
table
SELECT
 trnsact.sku
 ,skuinfo.vendor
FROM trnsact LEFT JOIN skuinfo
ON trnsact.sku=skuinfo.sku
GROUP BY trnsact.sku, skuinfo.vendor

## STEP 2 - joining the second table and counting absent skus,
grouping by vendor and ordering by #absent skus
SELECT
  COUNT(DISTINCT skuinfo_trnsact.sku) AS absent_from_skstinfo
  , skuinfo_trnsact.vendor
FROM (SELECT
 trnsact.sku
 ,skuinfo.vendor
FROM trnsact LEFT JOIN skuinfo
ON trnsact.sku=skuinfo.sku
GROUP BY trnsact.sku, skuinfo.vendor) AS skuinfo_trnsact LEFT JOIN
skstinfo
ON skuinfo_trnsact.sku=skstinfo.sku
WHERE skstinfo.sku IS NULL
GROUP BY skuinfo_trnsact.vendor
ORDER BY absent_from_skstinfo DESC
```

```
**********************************************************************
**
EXERCISE 6 - What is the brand of the sku with the greatest standard
deviation in sprice? Only examine skus which have been part of over
100 transactions.
**********************************************************************
**

SELECT DISTINCT top10.sku, top10.brand, top10.std_sprice
FROM (
SELECT TOP 10
 t.sku
 , s.brand
 , STDDEV_SAMP(t.sprice) AS std_sprice
 , COUNT(t.sku) AS num_sku
FROM trnsact t JOIN skuinfo s
ON t.sku=s.sku
GROUP BY t.sku, s.brand
HAVING num_sku>=100
ORDER BY std_sprice DESC) AS top10


**********************************************************************
**
EXERCISE 7 - What is the city and state of the store which had the
greatest increase in average daily revenue (as defined in Teradata
Week 5 Exercise Guide) from November to December?
**********************************************************************
**

SELECT
  (seasonal.nextmonth_sales-seasonal.month_sales) AS seasonal_trend
  , seasonal.city
  , seasonal.store
  , seasonal.state
FROM (SELECT
   aggregator.store
   , aggregator.city
   , aggregator.state
   , SUM(CASE WHEN aggregator.month_num=11 THEN
aggregator.total_revenue/aggregator.date_num END) AS month_sales
   , SUM(CASE WHEN aggregator.month_num=12 THEN
aggregator.total_revenue/aggregator.date_num END) AS nextmonth_sales
FROM (SELECT
  trnsact.store
  , store_msa.city
  , store_msa.state
  , COUNT(DISTINCT saledate) AS date_num
  , EXTRACT(YEAR from trnsact.saledate) as year_num
```

```
  , EXTRACT(MONTH from trnsact.saledate) as month_num
  , SUM(trnsact.amt) AS total_revenue
  , TRIM(EXTRACT(YEAR from trnsact.saledate))||TRIM(EXTRACT(MONTH from
trnsact.saledate)) AS yearmonth
  FROM trnsact JOIN store_msa
  ON trnsact.store=store_msa.store
  GROUP BY yearmonth, trnsact.store, store_msa.city, store_msa.state,
year_num, month_num
  WHERE trnsact.stype='P' AND yearmonth<>'20058'
  HAVING date_num>=20) AS aggregator
GROUP BY aggregator.store, aggregator.city, aggregator.state) AS
seasonal
ORDER BY seasonal_trend DESC;


**************************************************************************
*****
EXERCISE 8 - Compare the average daily revenue (as defined in Teradata
Week 5 Exercise Guide) of the store with the highest msa_income and
the store with the lowest median msa_income (according to the
msa_income field). In what city and state were these two stores, and
which store had a higher average daily revenue?
**************************************************************************
*****

SELECT TOP 10
  cleaned_sales_per_day.store
  , cleaned_sales_per_day.sales_per_day
  , store_msa.msa_income
  , store_msa.city
  , store_msa.state
FROM (SELECT
  EXTRACT(YEAR from saledate) as year_num
  , EXTRACT(MONTH from saledate) as month_num
  , store
  , SUM(trnsact.amt)/COUNT(DISTINCT saledate) AS sales_per_day
  , TRIM(EXTRACT(YEAR from saledate))||TRIM(EXTRACT(MONTH from
saledate)) AS yearmonth
  FROM trnsact
  GROUP BY month_num, year_num, store
  WHERE stype='P' AND yearmonth <> '20058'
  HAVING COUNT(DISTINCT saledate)>=20) AS cleaned_sales_per_day
LEFT JOIN store_msa
ON cleaned_sales_per_day.store=store_msa.store
ORDER BY store_msa.msa_income ASC;


**************************************************************************
***
EXERCISE 9 - Divide the msa_income groups up so that msa_incomes
```

```
between 1 and 20,000 are labeled 'low', msa_incomes between 20,001 and
30,000 are labeled 'med-low', msa_incomes between 30,001 and 40,000
are labeled 'med-high', and msa_incomes between 40,001 and 60,000 are
labeled 'high'. Which of these groups has the highest average daily
revenue (as defined in Teradata Week 5 Exercise Guide) per store?
**********************************************************************
***

SELECT

SUM(cleaned_sales_per_day.sales_per_day*cleaned_sales_per_day.date_num
)/SUM(cleaned_sales_per_day.date_num)
  , income_store.income_group
FROM (SELECT
  aggregated.year_num,
  aggregated.month_num,
  aggregated.date_num,
  aggregated.store,
  aggregated.sales_per_day,
  aggregated.yearmonth
FROM (SELECT
  EXTRACT(YEAR from saledate) as year_num,
  EXTRACT(MONTH from saledate) as month_num,
  COUNT(DISTINCT EXTRACT(DAY from saledate)) as date_num,
  store,
  SUM(trnsact.amt)/COUNT(DISTINCT EXTRACT(DAY from saledate)) AS
sales_per_day,
  TRIM(EXTRACT(YEAR from saledate))||TRIM(EXTRACT(MONTH from
saledate)) AS yearmonth
FROM trnsact
GROUP BY month_num, year_num, store
WHERE stype='P') AS aggregated
WHERE aggregated.date_num>=20 AND aggregated.yearmonth <> '20058') AS
cleaned_sales_per_day
LEFT JOIN
  (SELECT store, (CASE
  WHEN msa_income >1 AND msa_income <= 20000 THEN 'low'
  WHEN msa_income > 20000 AND msa_income <= 30000 THEN 'med-low'
  WHEN msa_income > 30000 AND msa_income <= 40000 THEN 'med-high'
  WHEN msa_income > 40000 AND msa_income <= 60000 THEN 'high' END) AS
income_group
  FROM store_msa) AS income_store
ON cleaned_sales_per_day.store=income_store.store
GROUP BY income_store.income_group;

### could also use BETWEEN function in CASE statement ###

**********************************************************************
*****
EXERCISE 10 - Divide stores up so that stores with msa populations
```

between 1 and 100,000 are labeled 'very small', stores with msa populations between 100,001 and 200,000 are labeled 'small', stores with msa populations between 200,001 and 500,000 are labeled 'med_small', stores with msa populations between 500,001 and 1,000,000 are labeled 'med_large', stores with msa populations between 1,000,001 and 5,000,000 are labeled “large”, and stores with msa_population greater than 5,000,000 are labeled “very large”. What is the average daily revenue (as defined in Teradata Week 5 Exercise Guide) for a store in a “very large” population msa?
****

```
SELECT

SUM(cleaned_sales_per_day.sales_per_day*cleaned_sales_per_day.date_num
)/SUM(cleaned_sales_per_day.date_num)
  , msa_store.pop_size
FROM (SELECT
  aggregated.year_num,
  aggregated.month_num,
  aggregated.date_num,
  aggregated.store,
  aggregated.sales_per_day,
  aggregated.yearmonth
FROM (SELECT
  EXTRACT(YEAR from saledate) as year_num,
  EXTRACT(MONTH from saledate) as month_num,
  COUNT(DISTINCT EXTRACT(DAY from saledate)) as date_num,
  store,
  SUM(trnsact.amt)/COUNT(DISTINCT EXTRACT(DAY from saledate)) AS
sales_per_day,
  TRIM(EXTRACT(YEAR from saledate))||TRIM(EXTRACT(MONTH from
saledate)) AS yearmonth
FROM trnsact
GROUP BY month_num, year_num, store
WHERE stype='P') AS aggregated
WHERE aggregated.date_num>=20 AND aggregated.yearmonth <> '20058') AS
cleaned_sales_per_day LEFT JOIN (SELECT store, (CASE
  WHEN msa_pop >1 AND msa_pop <= 100000 THEN 'very small'
  WHEN msa_pop > 100000 AND msa_pop <= 200000 THEN 'small'
  WHEN msa_pop > 200000 AND msa_pop <= 500000 THEN 'med-small'
  WHEN msa_pop > 500000 AND msa_pop <= 1000000 THEN 'med-large'
  WHEN msa_pop > 1000000 AND msa_pop <= 5000000 THEN 'large'
  WHEN msa_pop > 5000000 THEN 'very large' END) AS pop_size
FROM store_msa) AS msa_store
ON cleaned_sales_per_day.store=msa_store.store
GROUP BY msa_store.pop_size;
```

****

```
****
EXERCISE 11 - Which department in which store had the greatest percent
increase in average daily sales revenue from November to December, and
what city and state was that store located in? Only examine
departments whose total sales were at least $1,000 in both November
and December.
**********************************************************************
****
SELECT
  ((seasonal.nextmonth_sales-seasonal.month_sales)/
seasonal.month_sales*100) AS seasonal_trend
  , seasonal.city
  , seasonal.store
  , seasonal.state
  , seasonal.dept
FROM (SELECT
   aggregator.city
   , aggregator.store
   , aggregator.state
   , aggregator.dept
   , SUM(CASE WHEN aggregator.month_num=11 THEN
aggregator.total_revenue/aggregator.date_num END) AS month_sales
   , SUM(CASE WHEN aggregator.month_num=12 THEN
aggregator.total_revenue/aggregator.date_num END) AS nextmonth_sales
   , SUM(CASE WHEN aggregator.month_num=11 OR aggregator.month_num=12
THEN aggregator.total_revenue END) AS sales_Nov_Dec
FROM (SELECT
  trnsact.store
  , store_msa.city
  , store_msa.state
  , skuinfo.dept
  , EXTRACT(YEAR from trnsact.saledate) as year_num
  , EXTRACT(MONTH from trnsact.saledate) as month_num
  , COUNT(DISTINCT saledate) as date_num
  , SUM(trnsact.amt) AS total_revenue
  , TRIM(EXTRACT(YEAR from trnsact.saledate))||TRIM(EXTRACT(MONTH from
trnsact.saledate)) AS yearmonth
  FROM (trnsact JOIN store_msa
  ON trnsact.store=store_msa.store) JOIN skuinfo
  ON trnsact.sku=skuinfo.sku
  GROUP BY yearmonth, trnsact.store, store_msa.city, store_msa.state,
year_num, month_num, skuinfo.dept
  WHERE trnsact.stype='P' AND yearmonth<>'20058'
  HAVING date_num>=20) AS aggregator
HAVING sales_Nov_Dec >= 1000
GROUP BY aggregator.store, aggregator.city, aggregator.state,
aggregator.dept) AS seasonal
ORDER BY seasonal_trend DESC;
```

```
**************************************************************************
****
EXERCISE 12 - Which department within a particular store had the
greatest decrease in average daily sales revenue from August to
September, and in what city and state was that store located?
**************************************************************************
****

SELECT
  (seasonal.nextmonth_sales-seasonal.month_sales) AS seasonal_trend
  , seasonal.city
  , seasonal.store
  , seasonal.state
  , seasonal.dept
FROM (SELECT
   aggregator.city
   , aggregator.store
   , aggregator.state
   , aggregator.dept
   , SUM(CASE WHEN aggregator.month_num=8 THEN
aggregator.total_revenue/aggregator.date_num END) AS month_sales
   , SUM(CASE WHEN aggregator.month_num=9 THEN
aggregator.total_revenue/aggregator.date_num END) AS nextmonth_sales
FROM (SELECT
  trnsact.store
  , store_msa.city
  , store_msa.state
  , skuinfo.dept
  , EXTRACT(YEAR from trnsact.saledate) as year_num
  , EXTRACT(MONTH from trnsact.saledate) as month_num
  , COUNT(DISTINCT trnsact.saledate) as date_num
  , SUM(trnsact.amt) AS total_revenue
  , TRIM(EXTRACT(YEAR from trnsact.saledate))||TRIM(EXTRACT(MONTH from
trnsact.saledate)) AS yearmonth
  FROM (trnsact JOIN store_msa
  ON trnsact.store=store_msa.store) JOIN skuinfo
  ON trnsact.sku=skuinfo.sku
  GROUP BY yearmonth, trnsact.store, store_msa.city, store_msa.state,
year_num, month_num, skuinfo.dept
  WHERE trnsact.stype='P' AND yearmonth<>'20058'
  HAVING date_num>=20) AS aggregator
GROUP BY aggregator.store, aggregator.city, aggregator.state,
aggregator.dept) AS seasonal
WHERE seasonal_trend IS NOT NULL
ORDER BY seasonal_trend ASC;


**************************************************************************
****
EXERCISE 13 - Identify which department, in which city and state of
```

what store, had the greatest DECREASE in the number of items sold from August to September. How many fewer items did that department sell in September compared to August?
****

```
SELECT
  (seasonal.nextmonth_sales-seasonal.month_sales) AS seasonal_trend
  , seasonal.city
  , seasonal.store
  , seasonal.state
  , seasonal.dept
FROM (SELECT
   aggregator.city
   , aggregator.store
   , aggregator.state
   , aggregator.dept
   , SUM(CASE WHEN aggregator.month_num=8 THEN aggregator.items_sold
END) AS month_sales
   , SUM(CASE WHEN aggregator.month_num=9 THEN aggregator.items_sold
END) AS nextmonth_sales
FROM (SELECT
  trnsact.store
  , store_msa.city
  , store_msa.state
  , skuinfo.dept
  , EXTRACT(YEAR from trnsact.saledate) as year_num
  , EXTRACT(MONTH from trnsact.saledate) as month_num
  , COUNT(DISTINCT trnsact.saledate) as date_num
  , COUNT (trnsact.sku) AS items_sold
  , TRIM(EXTRACT(YEAR from trnsact.saledate))||TRIM(EXTRACT(MONTH from
trnsact.saledate)) AS yearmonth
  FROM (trnsact JOIN store_msa
  ON trnsact.store=store_msa.store) JOIN skuinfo
  ON trnsact.sku=skuinfo.sku
  GROUP BY yearmonth, trnsact.store, store_msa.city, store_msa.state,
year_num, month_num, skuinfo.dept
  WHERE trnsact.stype='P' AND yearmonth<>'20058'
  HAVING date_num>=20) AS aggregator
GROUP BY aggregator.store, aggregator.city, aggregator.state,
aggregator.dept) AS seasonal
WHERE seasonal_trend IS NOT NULL
ORDER BY seasonal_trend ASC;
```

****
EXERCISE 14 - For each store, determine the month with the minimum average daily revenue (as defined in Teradata Week 5 Exercise Guide) . For each of the twelve months of the year, count how many stores'

```
minimum average daily revenue was in that month. During which month(s)
did over 100 stores have their minimum average daily revenue?
**********************************************************************
******

SELECT
  COUNT(ranked.store)
  ,ranked.month_num
FROM (SELECT
  aggregated.year_num
  ,aggregated.month_num
  ,aggregated.date_num
  ,aggregated.store
  ,aggregated.sales_per_day
  ,aggregated.yearmonth
  ,ROW_NUMBER() OVER (PARTITION BY aggregated.store ORDER BY
aggregated.sales_per_day ASC) AS store_rownum
FROM (SELECT
  EXTRACT(YEAR from saledate) as year_num,
  EXTRACT(MONTH from saledate) as month_num,
  COUNT(DISTINCT saledate) as date_num,
  store,
  SUM(trnsact.amt)/COUNT(DISTINCT EXTRACT(DAY from saledate)) AS
sales_per_day,
  TRIM(EXTRACT(YEAR from saledate))||TRIM(EXTRACT(MONTH from
saledate)) AS yearmonth
FROM trnsact
GROUP BY month_num, year_num, store
WHERE stype='P') AS aggregated
WHERE aggregated.date_num>=20 AND aggregated.yearmonth <> '20058'
QUALIFY store_rownum=1) AS ranked
GROUP BY ranked.month_num;


**********************************************************************
****
EXERCISE 15 - Write a query that determines the month in which each
store had its maximum number of sku units returned. During which month
did the greatest number of stores have their maximum number of sku
units returned?
**********************************************************************
****

SELECT
  COUNT(ranked.store)
  ,ranked.month_num
FROM (SELECT
  aggregated.year_num
  ,aggregated.month_num
  ,aggregated.store
```

```
  ,aggregated.num_items_returned
  ,ROW_NUMBER() OVER (PARTITION BY aggregated.store ORDER BY
aggregated.num_items_returned DESC) AS store_rownum
FROM (SELECT
  EXTRACT(YEAR from saledate) as year_num,
  EXTRACT(MONTH from saledate) as month_num,
  COUNT(DISTINCT saledate) as date_num,
  store,
  COUNT(DISTINCT trnsact.sku) AS num_items_returned,
  TRIM(EXTRACT(YEAR from saledate))||TRIM(EXTRACT(MONTH from
saledate)) AS yearmonth
FROM trnsact
GROUP BY month_num, year_num, store
WHERE stype='R') AS aggregated
WHERE aggregated.date_num>=20 AND aggregated.yearmonth <> '20058'
QUALIFY store_rownum=1) AS ranked
GROUP BY ranked.month_num;
```