-- #1 For each business, compute the moving average of the costs for the past thirty days.

SELECT \*,

-- Compute the moving average of spending

(SELECT AVG(s2.spending) FROM ads\_spending s2

WHERE (s1.date - s2.date) <= 30 AND

(s1.date - s2.date) >= 0 AND

s1.business = s2.business) AS moving\_avg

FROM ads\_spending s1;

-- #2 - For each business, compute the moving standard deviation of the costs for the past thirty days.

SELECT \*,

-- Compute the moving std of spending

(SELECT POWER(AVG(POWER(s3.spending - t1.moving\_avg, 2)), 0.5) FROM ads\_spending s3

WHERE (t1.date - s3.date) <= 30 AND

(t1.date - s3.date) >= 0 AND

t1.business = s3.business) AS moving\_std

FROM (

SELECT \*,

-- Compute the moving average of spending

(SELECT AVG(s2.spending) FROM ads\_spending s2

WHERE (s1.date - s2.date) <= 30 AND

(s1.date - s2.date) >= 0 AND

s1.business = s2.business) AS moving\_avg

FROM ads\_spending s1

) t1;

-- #3 - The platform wants to track anomalous spendings. Create a new column called â€œoutlierâ€

-- which flags any spending that is above or below the two standard deviation from the mean.

-- Use the moving average and standard deviation computed in previous steps.

SELECT \*,

-- Append a new column called â€˜outlierâ€™ that flags outlier spendings

(CASE WHEN spending > moving\_avg + 2 \* moving\_std THEN 1

WHEN spending < moving\_avg - 2 \* moving\_std THEN 1

else 0

END) AS outlier

FROM (

SELECT \*,

-- Compute the moving std of spending

(SELECT POWER(AVG(POWER(s3.spending - t1.moving\_avg, 2)), 0.5) FROM ads\_spending s3

WHERE (t1.date - s3.date) <= 30 AND

(t1.date - s3.date) >= 0 AND

t1.business = s3.business) AS moving\_std

FROM (

SELECT \*,

-- Compute the moving average of spending

(SELECT AVG(s2.spending) FROM ads\_spending s2

WHERE (s1.date - s2.date) <= 30 AND

(s1.date - s2.date) >= 0 AND

s1.business = s2.business) AS moving\_avg

FROM ads\_spending s1

) t1

) t2;