**Code:**

Profit etc by transaction - taking into account returns

# Join product\_costs so we can work out the cost to prism by transaction\_id

WITH with\_costs AS (

SELECT

      ti.\*,

      pc.cost\_of\_item

FROM `Prism\_Main.transactionsanditems` AS ti

JOIN `Prism\_Main.product\_costs` AS pc

  ON ti.item\_id = pc.item\_id)

,

# join with returns

with\_returns AS (

SELECT

      wc.\*,

      COALESCE(pr.return\_quantity, 0) AS return\_quantity,

      pr.return\_status

FROM with\_costs AS wc

LEFT JOIN `Prism\_Main.product\_returns` AS pr

  ON wc.transaction\_id = pr.transaction\_id

  AND wc.item\_id = pr.item\_id)

,

# quantity after returns (exchanges left the same)

quantities\_after\_returns AS (

SELECT

      \*,

      CASE

        WHEN return\_status = 'Refund' THEN item\_quantity - return\_quantity

        ELSE item\_quantity

        END AS quantity\_after\_return,

      CASE

        WHEN return\_status = 'Refund' THEN return\_quantity \* item\_price

        ELSE 0

        END AS refund\_amount

FROM with\_returns)

,

# calculating refund amount and quantities/prices/costs before and after returns by item

by\_item\_calcs AS (

SELECT

      transaction\_id,

      item\_id,

      item\_price,

      cost\_of\_item,

      item\_quantity,

      return\_quantity,

      return\_status,

      refund\_amount,

      quantity\_after\_return,

      item\_price \* item\_quantity AS price\_before\_returns,

      item\_price \* quantity\_after\_return AS price\_after\_returns,

      cost\_of\_item \* item\_quantity AS cost\_before\_return,

      cost\_of\_item \* quantity\_after\_return AS cost\_after\_returns

FROM quantities\_after\_returns)

,

# get totals per transaction

basket\_totals AS (

SELECT

      transaction\_id,

      SUM(item\_quantity) AS item\_quantity\_before\_return,

      SUM(refund\_amount) AS total\_refund,

      SUM(quantity\_after\_return) AS item\_quantity\_after\_return,

      SUM(price\_before\_returns) AS price\_before\_returns,

      SUM(price\_after\_returns) AS price\_after\_returns,

      SUM(cost\_before\_return) AS cost\_before\_return,

      SUM(cost\_after\_returns) AS cost\_after\_returns

FROM by\_item\_calcs

GROUP BY transaction\_id)

,

# Join above with transactions

# I've left some extra columns in just in case we want to use them later

transaction\_totals AS (

SELECT

      t.date,

      EXTRACT(YEAR FROM t.date) as year,

      EXTRACT(QUARTER FROM t.date) as quarter,

      EXTRACT(MONTH FROM t.date) as month,

      t.transaction\_id,

      t.transaction\_coupon,

      t.transaction\_revenue,

      t.transaction\_shipping,

      t.transaction\_total,

      bt.item\_quantity\_before\_return,

      bt.total\_refund,

      bt.item\_quantity\_after\_return,

      t.transaction\_revenue - bt.total\_refund AS transaction\_revenue\_after\_refund,

      bt.price\_before\_returns,

      bt.price\_after\_returns,

      bt.cost\_before\_return,

      bt.cost\_after\_returns

FROM `Prism\_Main.transactions` as t

LEFT JOIN basket\_totals as bt

  ON t.transaction\_id = bt.transaction\_id)

# calculate total profit margin by year

# Change this last bit to any select statement to play around with the whole table

SELECT

      year,

      ((SUM(transaction\_revenue\_after\_refund) - SUM(cost\_after\_returns))/(SUM(transaction\_revenue\_after\_refund ))\*100) as profit\_margin

FROM transaction\_totals

GROUP BY 1

ORDER BY 1

Code for Specifc Item Category:

# Join product\_costs so we can work out the cost to prism by transaction\_id

WITH with\_costs AS (

SELECT

      ti.\*,

      pc.cost\_of\_item

FROM `Prism\_Main.transactionsanditems` AS ti

JOIN `Prism\_Main.product\_costs` AS pc

  ON ti.item\_id = pc.item\_id)

,

# join with returns

with\_returns AS (

SELECT

      wc.\*,

      COALESCE(pr.return\_quantity, 0) AS return\_quantity,

      pr.return\_status

FROM with\_costs AS wc

LEFT JOIN `Prism\_Main.product\_returns` AS pr

  ON wc.transaction\_id = pr.transaction\_id

  AND wc.item\_id = pr.item\_id)

,

#Adding Item detatils to products

full\_details as (

  select wr.\*,

  pa.item\_name,

  pa.item\_brand,

  pa.item\_main\_category,

  pa.item\_sub\_category

  from with\_returns as wr

  left join `Prism\_Main.productattributes` as pa

  on wr.item\_id = pa.item\_id

),

# quantity after returns (exchanges left the same)

quantities\_after\_returns AS (

SELECT

      \*,

      CASE

        WHEN return\_status = 'Refund' THEN item\_quantity - return\_quantity

        ELSE item\_quantity

        END AS quantity\_after\_return,

      CASE

        WHEN return\_status = 'Refund' THEN return\_quantity \* item\_price

        ELSE 0

        END AS refund\_amount

FROM full\_details)

,

# calculating refund amount and quantities/prices/costs before and after returns by item

by\_item\_calcs AS (

SELECT

      transaction\_id,

      date,

      item\_id,

      item\_name,

      item\_brand,

      item\_main\_category,

      item\_price,

      cost\_of\_item,

      item\_quantity,

      return\_quantity,

      return\_status,

      refund\_amount,

      quantity\_after\_return,

      item\_price \* item\_quantity AS price\_before\_returns,

      item\_price \* quantity\_after\_return AS price\_after\_returns,

      cost\_of\_item \* item\_quantity AS cost\_before\_return,

      cost\_of\_item \* quantity\_after\_return AS cost\_after\_returns

FROM quantities\_after\_returns)

## final table with profit margin metrics

SELECT

      EXTRACT(YEAR FROM date) as year,

      item\_main\_category,

      ((SUM(price\_after\_returns) - SUM(cost\_after\_returns))/(SUM(price\_after\_returns))\*100) as profit\_margin,

      (SUM(price\_after\_returns) - SUM(cost\_after\_returns)) as profit\_totals,

      sum(quantity\_after\_return) as total\_sales,

      sum(refund\_amount) as returned\_amount,

      sum(return\_quantity) as number\_of\_returns,

      ((sum(return\_quantity))/(sum(item\_quantity))) as return\_percentage

FROM by\_item\_calcs

group by 1, 2

order by 1 desc, 3 desc

limit 100