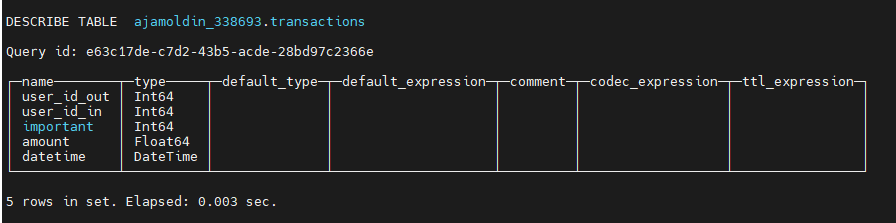
Short Report by Lab3.

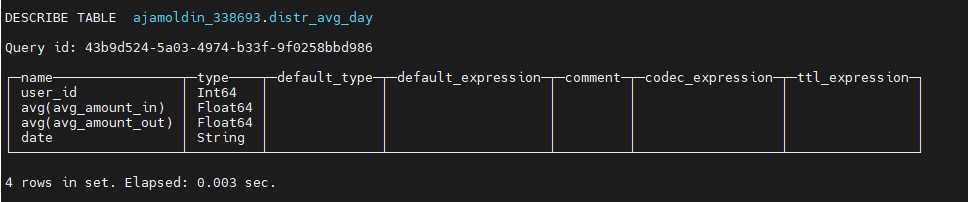
Yamoldin Alexander, J41321c

Table chemas

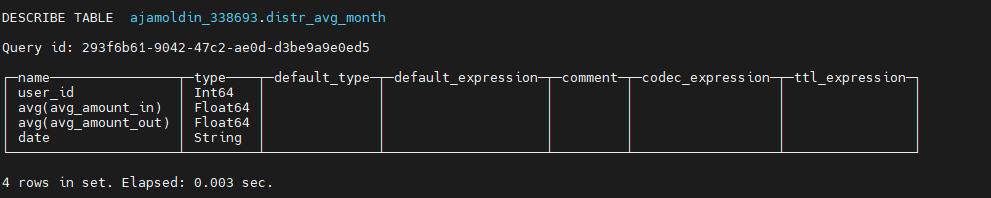
Transactions table



Average transactions per day



Average transactions per month



Task 1. Average amount for incoming and outcoming transactions by months and days for each user.

Average incoming transactions by day

CREATE MATERIALIZED VIEW ajamoldin\_338693.avg\_day ON CLUSTER kube\_clickhouse\_cluster

ENGINE = AggregatingMergeTree()

ORDER BY (user\_id, date)

POPULATE

AS SELECT

user\_id\_in AS user\_id,

avg(avg\_amount\_in),

avg(avg\_amount\_out),

date\_in AS date

FROM

(

SELECT

user\_id\_in,

avg(amount) AS avg\_amount\_in,

formatDateTime(datetime, '%d-%m-%G') AS date\_in

FROM ajamoldin\_338693.transactions

GROUP BY

user\_id\_in,

date\_in

) AS t1

INNER JOIN

(

SELECT

user\_id\_out,

avg(amount) AS avg\_amount\_out,

formatDateTime(datetime, '%d-%m-%G') AS date\_out

FROM ajamoldin\_338693.transactions

GROUP BY

user\_id\_out,

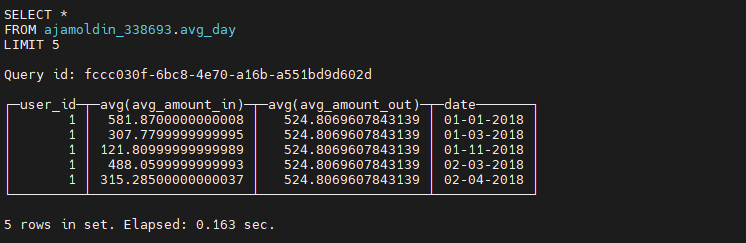
date\_out

) AS t2 ON t1.user\_id\_in = t2.user\_id\_out

GROUP BY date, user\_id

CREATE TABLE ajamoldin\_338693.distr\_avg\_day ON CLUSTER kube\_clickhouse\_cluster AS ajamoldin\_338693.avg\_day

ENGINE = Distributed(kube\_clickhouse\_cluster, ajamoldin\_338693, avg\_day, xxHash64(user\_id))



Average outcoming transactions by month

CREATE MATERIALIZED VIEW ajamoldin\_338693.avg\_month ON CLUSTER kube\_clickhouse\_cluster

ENGINE = AggregatingMergeTree()

ORDER BY (user\_id, date)

POPULATE

AS SELECT

user\_id\_in AS user\_id,

avg(avg\_amount\_in),

avg(avg\_amount\_out),

date\_in AS date

FROM

(

SELECT

user\_id\_in,

avg(amount) AS avg\_amount\_in,

formatDateTime(datetime, '%m-%G') AS date\_in

FROM ajamoldin\_338693.transactions

GROUP BY

user\_id\_in,

date\_in

) AS t1

INNER JOIN

(

SELECT

user\_id\_out,

avg(amount) AS avg\_amount\_out,

formatDateTime(datetime, '%m-%G') AS date\_out

FROM ajamoldin\_338693.transactions

GROUP BY

user\_id\_out,

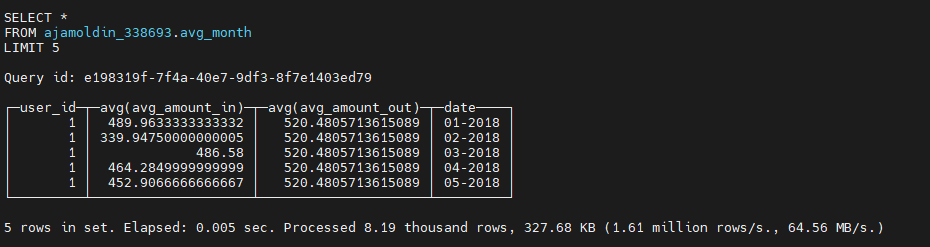
date\_out

) AS t2 ON t1.user\_id\_in = t2.user\_id\_out

GROUP BY date, user\_id

CREATE TABLE ajamoldin\_338693.distr\_avg\_month ON CLUSTER kube\_clickhouse\_cluster AS ajamoldin\_338693.avg\_month

ENGINE = Distributed(kube\_clickhouse\_cluster, ajamoldin\_338693, avg\_month, xxHash64(user\_id))



Task 3. The sums for incoming and outcoming transactions by months for each user.

CREATE MATERIALIZED VIEW ajamoldin\_338693.sum\_month ON CLUSTER kube\_clickhouse\_cluster

ENGINE = AggregatingMergeTree()

ORDER BY (user\_id, date)

POPULATE

AS SELECT

user\_id\_in AS user\_id,

sum(sum\_amount\_in),

sum(sum\_amount\_out),

date\_in AS date

FROM

(

SELECT

user\_id\_in,

sum(amount) AS sum\_amount\_in,

formatDateTime(datetime, '%m-%G') AS date\_in

FROM ajamoldin\_338693.transactions

GROUP BY

user\_id\_in,

date\_in

) AS t1

INNER JOIN

(

SELECT

user\_id\_out,

sum(amount) AS sum\_amount\_out,

formatDateTime(datetime, '%m-%G') AS date\_out

FROM ajamoldin\_338693.transactions

GROUP BY

user\_id\_out,

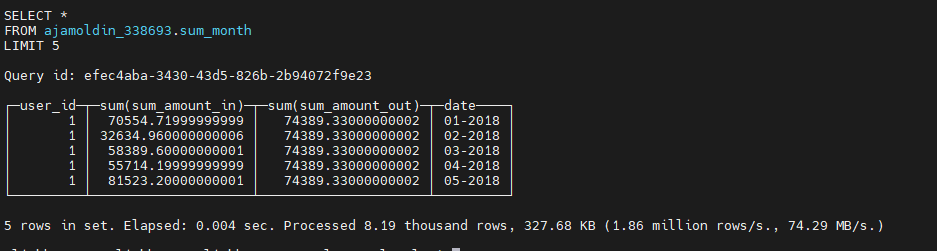
date\_out

) AS t2 ON t1.user\_id\_in = t2.user\_id\_out

GROUP BY date, user\_id

CREATE TABLE ajamoldin\_338693.distr\_sum\_month ON CLUSTER kube\_clickhouse\_cluster AS ajamoldin\_338693.sum\_month

ENGINE = Distributed(kube\_clickhouse\_cluster, ajamoldin\_338693, sum\_month, xxHash64(user\_id))



List of MV and creation queries and Sharding expression

We use shards by date to provide maximally loyalty of the data in mv evaluating. So in each node we have relevant data group by date.

