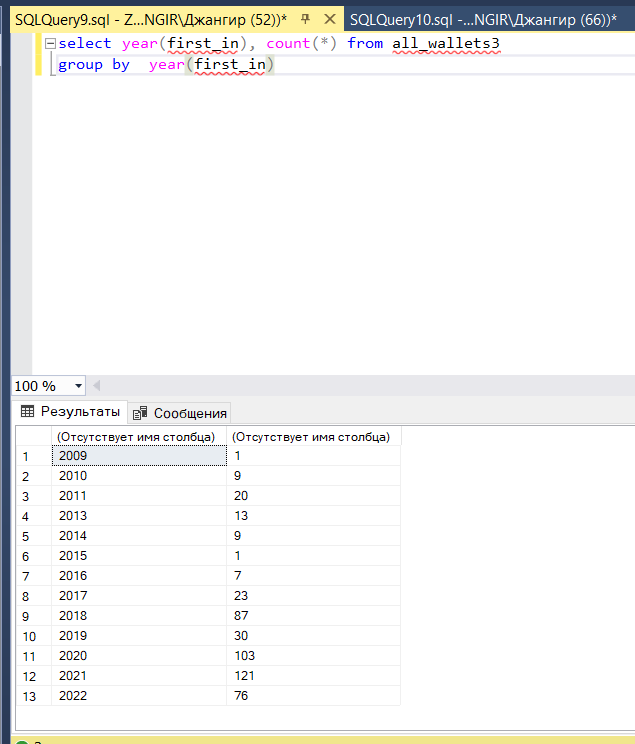
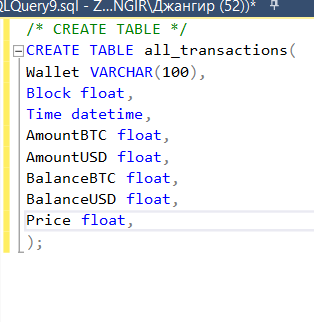


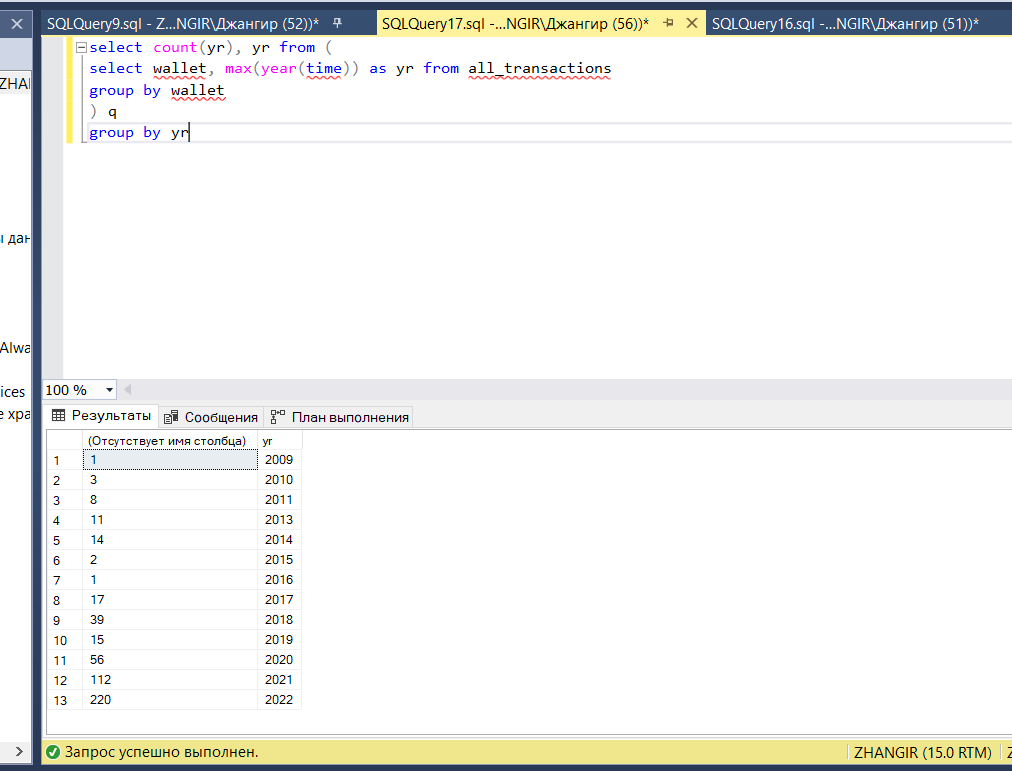
В руках первых 500 людей находится 28 процентов от общего кол-ва на руках.



Первое пополнение кошелька



Удалил Primary key на block, так как в этом поле есть дупликаты



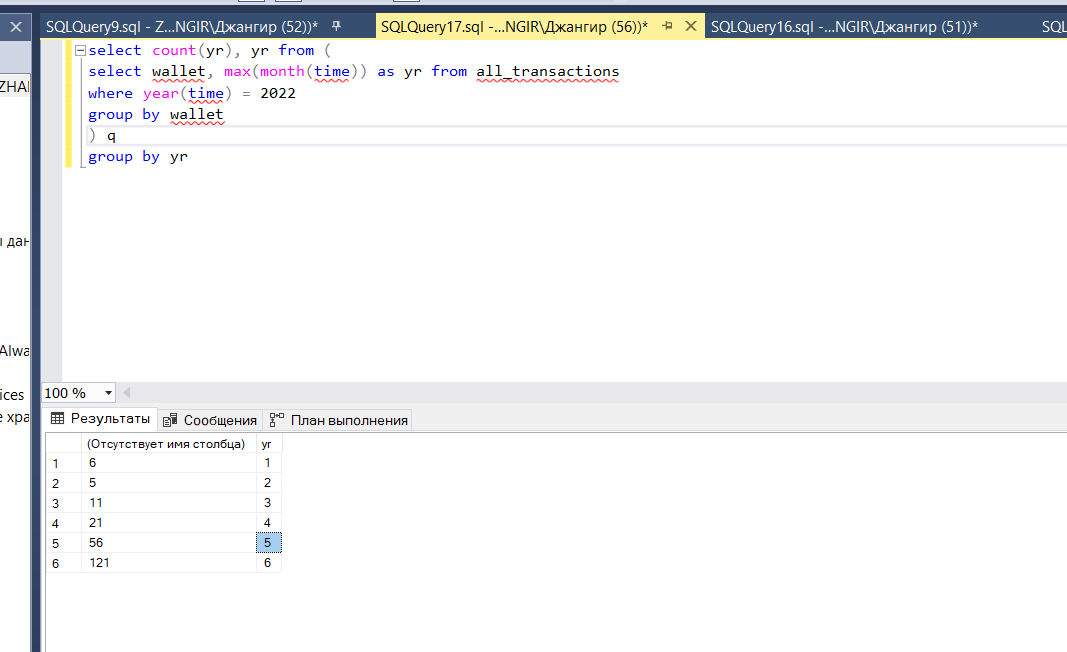
Последние операции с кошельков по годам

44% процента делали в этом году

66,4% процента делали как минимум с 2021 года (22,4)

77,6% процента делали как минимум с 2020 года (10,8)

8% процентов не делали уже более 5 лет



select count(yr), yr from (

select wallet, max(month(time)) as yr from all\_transactions

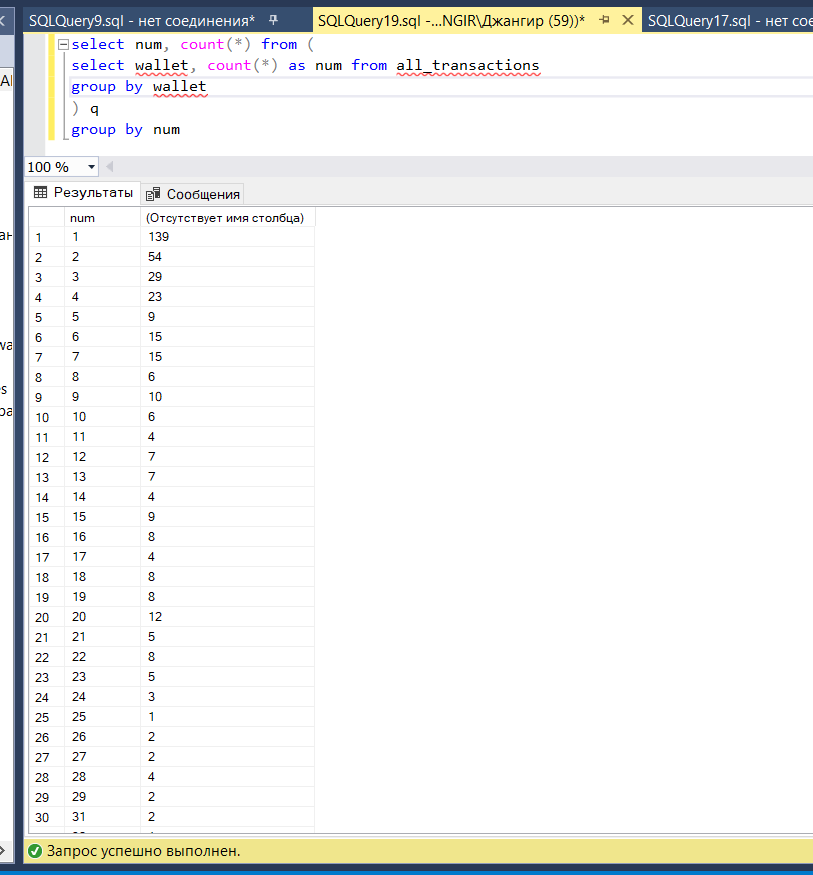
where year(time) = 2022

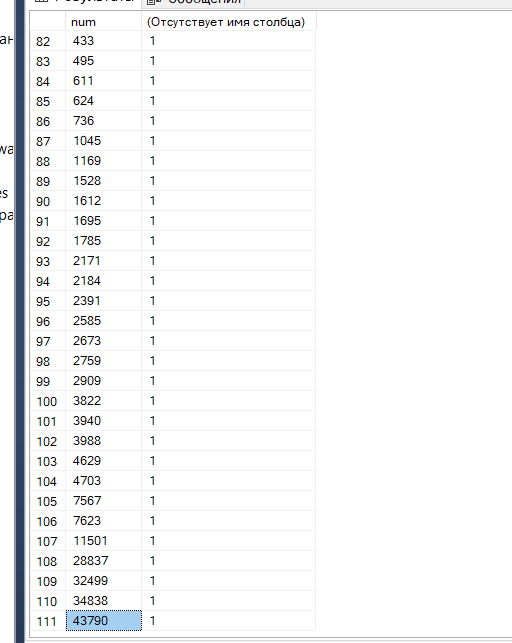
group by wallet

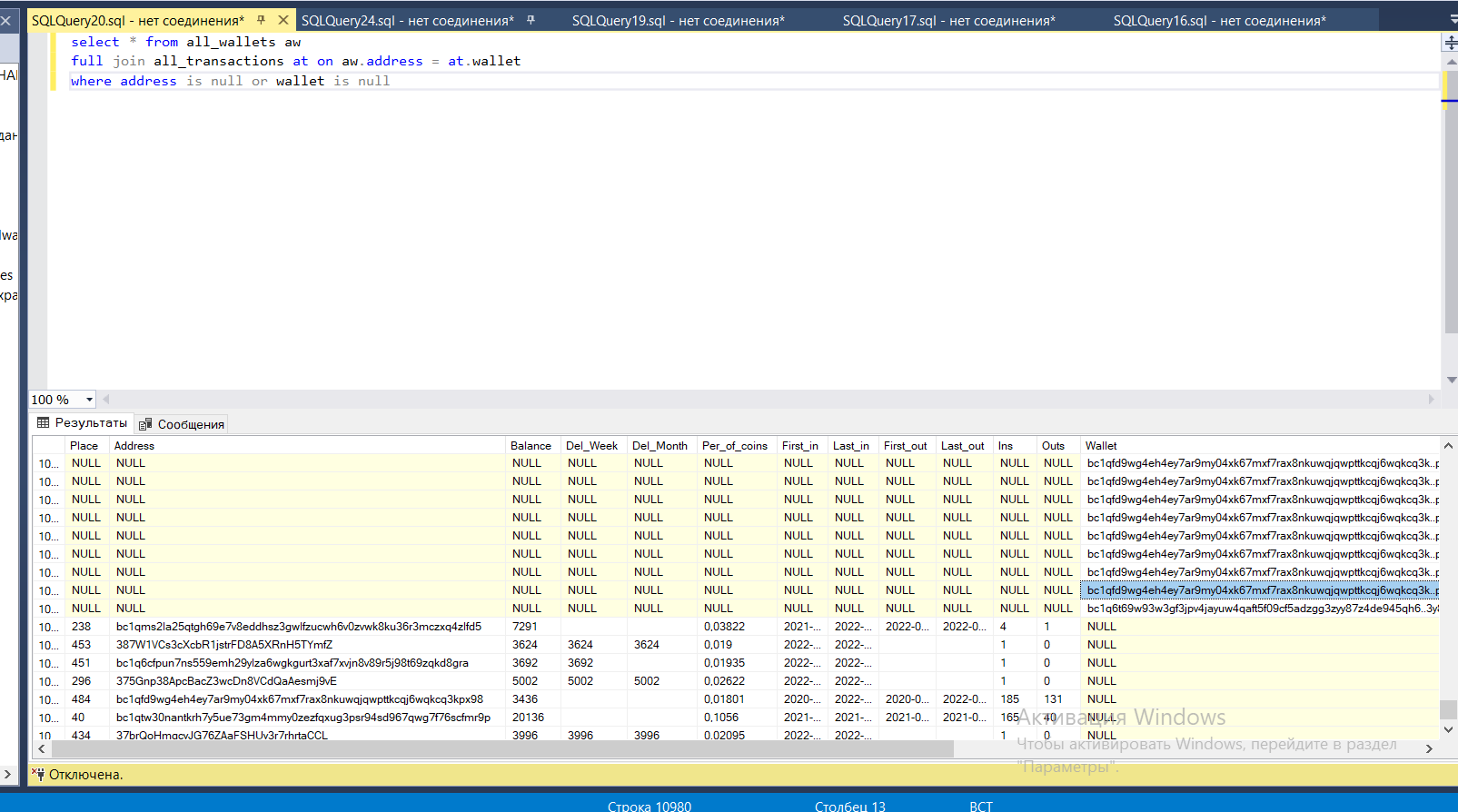
) q

group by yr

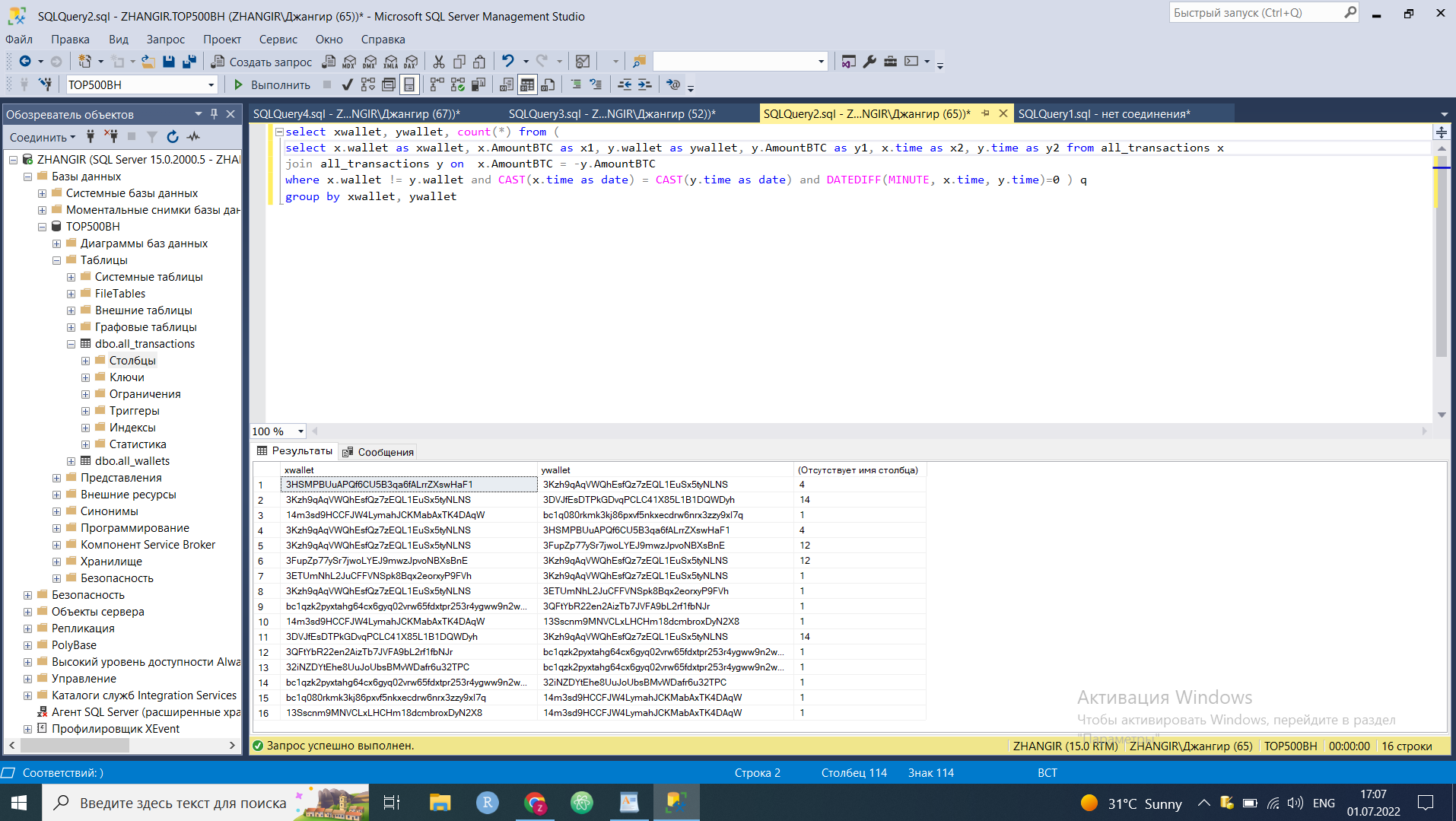
Последние операции с кошельков по 2022 году (месяцам)







При парсинге взялись не одни и те же люди ( ПРИ РАЗНЫХ ПАРСИНГАХ В РАЗНОЕ ВРЕМЯ СПИСОК ЛЮДЕЙ ПОМЕНЯЛСЯ )



select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.95 < y.AmountBTC

where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0--) q

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select xwallet, ywallet, count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

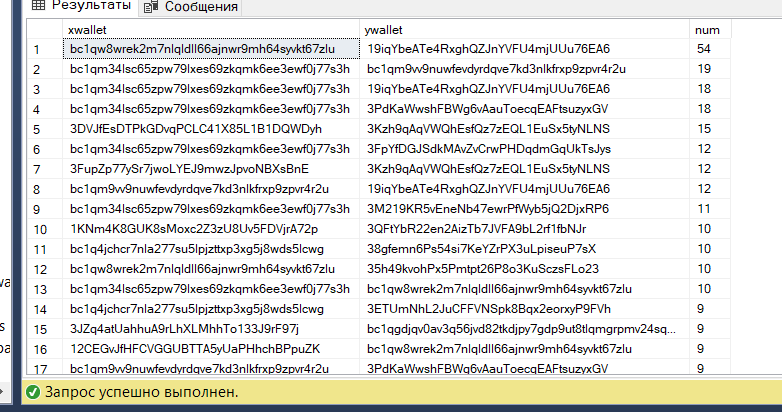
join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.95 < y.AmountBTC

where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0 ) q

group by xwallet, ywallet

order by num desc

REsults:



select m.\*, x.numoft, y.numoft from (

select xwallet, ywallet, count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.95 < y.AmountBTC

where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0 ) q

group by xwallet, ywallet ) m

--order by num desc

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) x on x.wallet = m.xwallet

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) y on y.wallet = m.ywallet

where x.numoft<100 or y.numoft<100

order by num desc

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select xwallet, count(\*) as num from (

select xwallet, ywallet, count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.95 < y.AmountBTC

where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0) q

group by xwallet, ywallet ) qq

group by xwallet

order by num desc

­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select m.\*, x.numoft, xx.first\_in from (

select xwallet, count(\*) as num from (

select xwallet, ywallet, count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.95 < y.AmountBTC

where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0) q

group by xwallet, ywallet ) qq

group by xwallet ) m

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) x on x.wallet = m.xwallet

JOIN (

select wallet, first\_in from all\_wallets

group by wallet, first\_in ) xx on xx.wallet = m.xwallet

order by num desc

СВЯЗАННЫЕ КОШЕЛЬКИ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

СВЯЗАННЫЕ КОШЕЛЬКИ 2 ( похожие транзакции )

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)<2 and DATEDIFF(minute, x.time, y.time)>-2

--(x.AmountBTC >= y.AmountBTC and x.AmountBTC\*0.9 <= y.AmountBTC ) or (y.AmountBTC >= x.AmountBTC and y.AmountBTC\*0.9 <= x.AmountBTC )

where x.wallet != y.wallet and ((x.AmountBTC>0 and y.AmountBTC>0) or (x.AmountBTC<0 and y.AmountBTC<0))

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select m.\*, x.numoft, y.numoft from (

select xwallet, ywallet, count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)<2 and DATEDIFF(minute, x.time, y.time)>-2

--(x.AmountBTC >= y.AmountBTC and x.AmountBTC\*0.9 <= y.AmountBTC ) or (y.AmountBTC >= x.AmountBTC and y.AmountBTC\*0.9 <= x.AmountBTC )

where x.wallet != y.wallet and ((x.AmountBTC>0 and y.AmountBTC>0) or (x.AmountBTC<0 and y.AmountBTC<0)) ) q

group by xwallet, ywallet ) m

--order by num desc

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) x on x.wallet = m.xwallet

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) y on y.wallet = m.ywallet

where num>5

order by num desc

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select m.\*, x.numoft, y.numoft from (

select xwallet, ywallet, count(\*) as num, avg((x1+y1)/2) as avgg from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)<2 and DATEDIFF(minute, x.time, y.time)>-2

--(x.AmountBTC >= y.AmountBTC and x.AmountBTC\*0.9 <= y.AmountBTC ) or (y.AmountBTC >= x.AmountBTC and y.AmountBTC\*0.9 <= x.AmountBTC )

where x.wallet != y.wallet and ((x.AmountBTC>0 and y.AmountBTC>0) or (x.AmountBTC<0 and y.AmountBTC<0)) ) q

group by xwallet, ywallet ) m

--order by num desc

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) x on x.wallet = m.xwallet

JOIN (

select wallet, count(\*) as numoft from all\_transactions

group by wallet ) y on y.wallet = m.ywallet

where num>100 or (avgg>100 or avgg<-100) and num>2

order by avgg desc

НУЖНО ЕЩЕ УЧИТЫВАТЬ ВРЕМЯ ПЕРВОГО ВЗНОСА

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ГРАФИКИ

WITH base AS

( SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions )

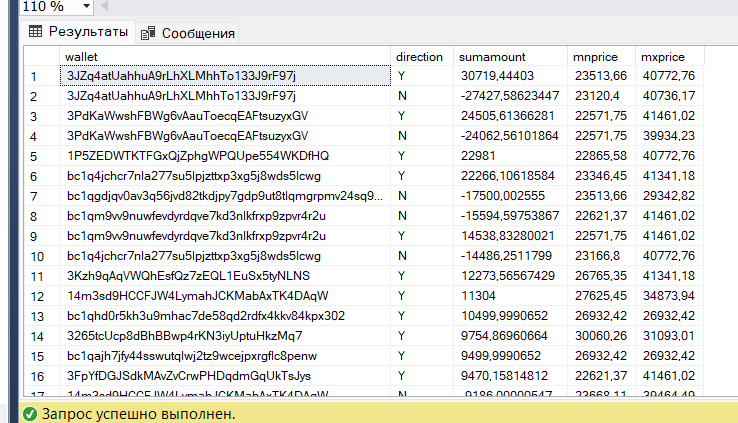
select wallet, direction, sum(amountBTC) as sumamount, min(price) as mnprice, max(price) as mxprice from base

where time between '2020-12-22' and '2021-01-11'

group by wallet, direction

ORDER BY abs(sum(amountBTC)) desc

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



много людей покупают по одной и той же цене в периоды дампов и апов

WITH basePOS AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' ),

baseNEG AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' )

select p1.\*, POSamountMinPrice FROM (

select wallet, min(price) as minprice, avg(price) as avgpriceMIN, count(\*) as trnum, sum(amountBTC) as POSamount from basePOS

where time between '2020-12-22' and '2021-01-11'

group by wallet

) p1

full join (

select wallet, sum(amountBTC) as POSamountMinPrice from basePOS x

where price <= 1.1\*

(select min(price) from basePOS y

where x.wallet= y.wallet

group by wallet)

group by wallet) p2 on p1.wallet = p2.wallet

order by POSamount desc

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

eei сделал сводку для одной даты для + и - трпанзакций

WITH basePOS AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2020-12-10' and '2021-01-11'),

baseNEG AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2020-12-10' and '2021-01-11')

SELECT P1.\*, '\_' as GAP11, N1.\*, CASE WHEN P1.WALLET IS NULL AND N1.Wallet IS NULL THEN 3 WHEN P1.WALLET IS NULL THEN 2 WHEN N1.Wallet IS NULL THEN 1 ELSE 0 END AS J FROM (

select p11.\*, POSamountMinPrice/POSamountP1\*100 as POSamountMinPrice1 FROM (

select wallet, min(price) as minprice1, avg(price) as avgpriceMIN1, count(\*) as trnumP1, sum(amountBTC) as POSamountP1, avg(amountBTC) as avgAmountP1 from basePOS

group by wallet

) p11

left join (

select wallet, sum(amountBTC) as POSamountMinPrice from basePOS x

where price <= 1.1\*

(select min(price) from basePOS y

where x.wallet= y.wallet )

group by wallet) p12 on p11.wallet = p12.wallet ) P1

FULL JOIN(

select n11.\*, POSamountMaxPrice/POSamountN1\*100 as POSamountMaxPrice1 FROM (

select wallet, max(price) as maxprice1, avg(price) as avgpriceN1, count(\*) as trnumN1, sum(amountBTC) as POSamountN1, avg(amountBTC) as avgAmountN1 from baseNEG

group by wallet

) n11

left join (

select wallet, sum(amountBTC) as POSamountMaxPrice from baseNEG x

where price >= 0.9\*

(select max(price) from baseNEG y

where x.wallet= y.wallet )

group by wallet) n12 on n11.wallet = n12.wallet ) N1 ON P1.Wallet = N1.Wallet

ORDER BY J

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

WITH basePOS1 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2020-12-10' and '2021-01-11'),

baseNEG1 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2020-12-10' and '2021-01-11'),

basePOS2 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-01-11' and '2021-01-23'),

baseNEG2 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-01-11' and '2021-01-23'),

basePOS3 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-01-27' and '2021-02-23'),

baseNEG3 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-01-27' and '2021-02-23'),

basePOS4 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-02-23' and '2021-03-02'),

baseNEG4 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-02-23' and '2021-03-02'),

basePOS5 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-03-02' and '2021-03-15'),

baseNEG5 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-03-02' and '2021-03-15'),

basePOS6 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-03-15' and '2021-03-27'),

baseNEG6 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-03-15' and '2021-03-27'),

basePOS7 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-05-08' and '2021-05-20'),

baseNEG7 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-05-08' and '2021-05-20'),

basePOS8 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-09-28' and '2021-10-22'),

baseNEG8 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-09-28' and '2021-10-22'),

basePOS9 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-11-07' and '2021-12-07'),

baseNEG9 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-11-07' and '2021-12-07'),

basePOS10 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-12-24' and '2022-01-24'),

baseNEG10 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-12-24' and '2022-01-24'),

basePOS11 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2022-05-03' and '2022-05-15'),

baseNEG11 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2022-05-03' and '2022-05-15'),

basePOS12 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2022-06-06' and '2022-06-20'),

baseNEG12 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2022-06-06' and '2022-06-20')

SELECT \* FROM (

SELECT rnk1+rnk2+rnk3+rnk4+rnk5+rnk6+rnk7+rnk8+rnk9+rnk10+rnk11+rnk12+rnk13+rnk14+rnk15+rnk16+rnk17+rnk18+rnk19+rnk20+rnk21+rnk22+rnk23+rnk24 as score\_yep ,\* FROM (

SELECT o.Wallet, COALESCE(rnk1, 0) as rnk1, COALESCE(rnk2,0) as rnk2 , COALESCE(rnk3,0) as rnk3, COALESCE(rnk4, 0) as rnk4, COALESCE(rnk5,0) as rnk5 , COALESCE(rnk6,0) as rnk6, COALESCE(rnk7, 0) as rnk7, COALESCE(rnk8,0) as rnk8 , COALESCE(rnk9,0) as rnk9, COALESCE(rnk10, 0) as rnk10, COALESCE(rnk11,0) as rnk11 , COALESCE(rnk12,0) as rnk12,

COALESCE(rnk13, 0) as rnk13, COALESCE(rnk14,0) as rnk14 , COALESCE(rnk15,0) as rnk15, COALESCE(rnk16, 0) as rnk16, COALESCE(rnk17,0) as rnk17 , COALESCE(rnk18,0) as rnk18, COALESCE(rnk19, 0) as rnk19, COALESCE(rnk20,0) as rnk20 , COALESCE(rnk21,0) as rnk21, COALESCE(rnk22, 0) as rnk22, COALESCE(rnk23,0) as rnk23 , COALESCE(rnk24,0) as rnk24 from

(SELECT DISTINCT wallet from all\_transactions) o

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk1 from (

select wallet, min(price) as minprice1 from basePOS1

group by wallet ) q ) P1 ON o.wallet = P1.wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk2 from (

select wallet, max(price) as maxprice1 from baseNEG1

group by wallet ) q ) N1 ON o.Wallet = N1.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk3 from (

select wallet, min(price) as minprice1 from basePOS2

group by wallet ) q ) P2 ON o.Wallet = P2.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk4 from (

select wallet, max(price) as maxprice1 from baseNEG2

group by wallet ) q ) N2 ON o.Wallet = N2.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk5 from (

select wallet, min(price) as minprice1 from basePOS3

group by wallet ) q ) P3 ON o.Wallet = P3.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk6 from (

select wallet, max(price) as maxprice1 from baseNEG3

group by wallet ) q ) N3 ON o.Wallet = N3.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk7 from (

select wallet, min(price) as minprice1 from basePOS4

group by wallet ) q ) P4 ON o.Wallet = P4.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk8 from (

select wallet, max(price) as maxprice1 from baseNEG4

group by wallet ) q ) N4 ON o.Wallet = N4.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk9 from (

select wallet, min(price) as minprice1 from basePOS5

group by wallet ) q ) P5 ON o.Wallet = P5.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk10 from (

select wallet, max(price) as maxprice1 from baseNEG5

group by wallet ) q ) N5 ON o.Wallet = N5.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk11 from (

select wallet, min(price) as maxprice1 from basePOS6

group by wallet ) q ) P6 ON o.Wallet = P6.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk12 from (

select wallet, max(price) as minprice1 from baseNEG6

group by wallet ) q ) N6 ON o.Wallet = N6.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk13 from (

select wallet, min(price) as maxprice1 from basePOS7

group by wallet ) q ) P7 ON o.Wallet = P7.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk14 from (

select wallet, max(price) as minprice1 from baseNEG7

group by wallet ) q ) N7 ON o.Wallet = N7.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk15 from (

select wallet, min(price) as maxprice1 from basePOS8

group by wallet ) q ) P8 ON o.Wallet = P8.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk16 from (

select wallet, max(price) as minprice1 from baseNEG8

group by wallet ) q ) N8 ON o.Wallet = N8.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk17 from (

select wallet, min(price) as maxprice1 from basePOS9

group by wallet ) q ) P9 ON o.Wallet = P9.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk18 from (

select wallet, max(price) as minprice1 from baseNEG9

group by wallet ) q ) N9 ON o.Wallet = N9.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk19 from (

select wallet, min(price) as maxprice1 from basePOS10

group by wallet ) q ) o0 ON o.Wallet = o0.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk20 from (

select wallet, max(price) as minprice1 from baseNEG10

group by wallet ) q ) N10 ON o.Wallet = N10.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk21 from (

select wallet, min(price) as maxprice1 from basePOS11

group by wallet ) q ) o1 ON o.Wallet = o1.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk22 from (

select wallet, max(price) as minprice1 from baseNEG11

group by wallet ) q ) N11 ON o.Wallet = N11.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by maxprice1) as rnk23 from (

select wallet, min(price) as maxprice1 from basePOS12

group by wallet ) q ) o2 ON o.Wallet = o2.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk24 from (

select wallet, max(price) as minprice1 from baseNEG12

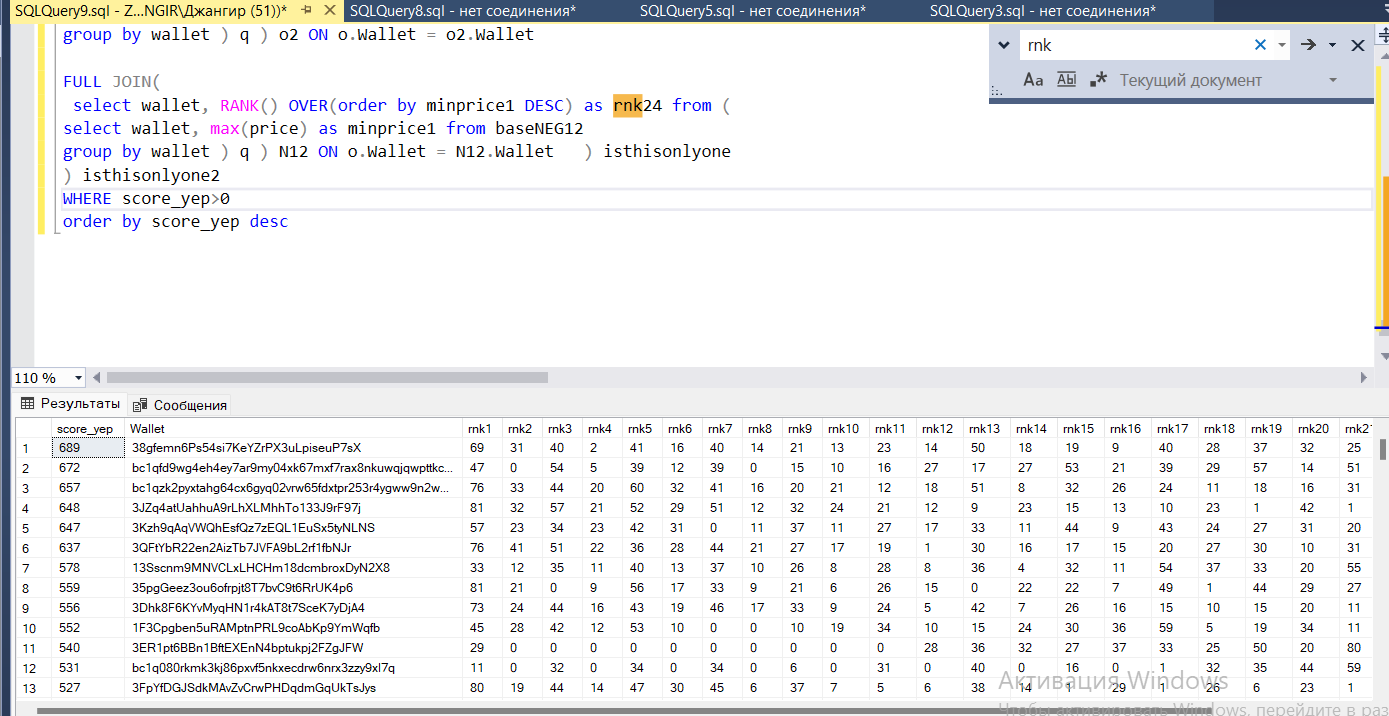
group by wallet ) q ) N12 ON o.Wallet = N12.Wallet ) isthisonlyone

) isthisonlyone2

WHERE score\_yep>0

order by score\_yep desc

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



Сводка для всех важных дат по кол-ву очков

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_select wallet, numL from (

select q1.wallet, CAST(num-num1 AS float)/num\*100 as numL from (

select wallet, num from (

select wallet, count(\*) as num from all\_transactions

group by wallet

) q

where num>2000

) q1

join (

select wallet, count(\*) as num1 from all\_transactions

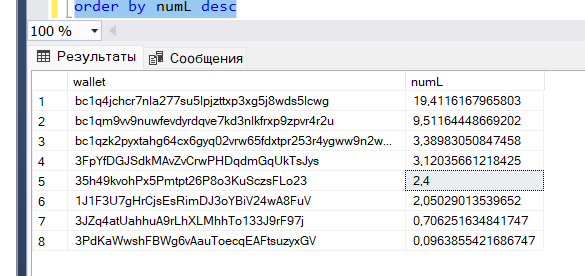
where AmountBTC>0.001 \* BalanceBTC or AmountBTC<-0.001 \* BalanceBTC

group by wallet

) q2 on q1.Wallet = q2.Wallet ) q3

where numL<20

order by numL desc



\_ интересные кошельки (СКОРЕЕ ВСЕГО ТРЕЙДЕРЫ)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

select wallet from (

select q1.wallet, CAST(num-num1 AS float)/num\*100 as numL from (

select wallet, num from (

select wallet, count(\*) as num from all\_transactions

group by wallet

) q

where num>2000

) q1

join (

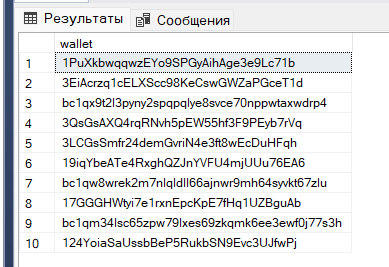
select wallet, count(\*) as num1 from all\_transactions

where AmountBTC>0.001 \* BalanceBTC or AmountBTC<-0.001 \* BalanceBTC

group by wallet

) q2 on q1.Wallet = q2.Wallet ) q3

where numL>20



кошельки которые мы не будем брать в расчет при дальнейшем изучении

SELECT xwallet, ywallet, avg(x1), avg(y1), count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.99 < y.AmountBTC

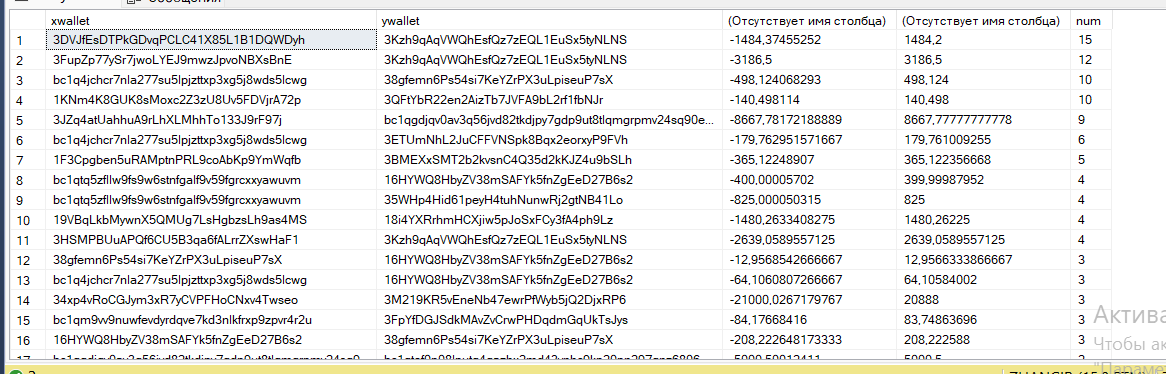
where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0

and x.AmountBTC<-0.1 and x.wallet not in ( select wallet from not\_counting) and y.wallet not in (select wallet from not\_counting)

) q

group by xwallet, ywallet

order by num desc



СВЯЗАННЫЕ КОШЕЛЬКИ (ПЕРЕБРОСЫ) версия 2.0

( убрали кошельки у которых очень много транзакций и которые составляют меньше 0.001 от их баланса )

( также поставили 0.99 разницу, так как считаю что комиссия не может быть больше 1%, либо это будут единичные случаи, которыми мы готовы пожертвовать ради чистоты данных )

( связь по времени поставили 0 секунд, опять же ради чистоты данных, ну и потому что разницы во времени> 0 минут совсем немного случаев )

( Очевидно будут попадаться случаи совпадения, но мы будем брать в среднем 3 и более соответствий для точности результатов )

select \* from (

SELECT xwallet, ywallet, avg(x1) as avgx1, avg(y1) avgy1, count(\*) as num from (

select x.wallet as xwallet, x.AmountBTC as x1, y.wallet as ywallet, y.AmountBTC as y1, x.time as x2, y.time as y2 from all\_transactions x

join all\_transactions y on -x.AmountBTC >= y.AmountBTC and -x.AmountBTC\*0.99 < y.AmountBTC

where x.wallet != y.wallet and CAST(x.time as date) = CAST(y.time as date) and DATEDIFF(minute, x.time, y.time)=0 and x.AmountBTC<0

and x.AmountBTC<-0.1 and x.wallet not in ( select wallet from not\_counting) and y.wallet not in (select wallet from not\_counting)

) q

group by xwallet, ywallet

) q2

-- where num >=3

order by num desc



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**MIN MAX PRICE ENJOYERЫ 2.0**

WITH basePOS1 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2020-12-10' and '2021-01-11'),

baseNEG1 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2020-12-10' and '2021-01-11'),

basePOS2 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-01-11' and '2021-01-23'),

baseNEG2 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-01-11' and '2021-01-23'),

basePOS3 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-01-27' and '2021-02-23'),

baseNEG3 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-01-27' and '2021-02-23'),

basePOS4 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-02-23' and '2021-03-02'),

baseNEG4 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-02-23' and '2021-03-02'),

basePOS5 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-03-02' and '2021-03-15'),

baseNEG5 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-03-02' and '2021-03-15'),

basePOS6 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-03-15' and '2021-03-27'),

baseNEG6 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-03-15' and '2021-03-27'),

basePOS7 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-05-08' and '2021-05-20'),

baseNEG7 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-05-08' and '2021-05-20'),

basePOS8 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-09-28' and '2021-10-22'),

baseNEG8 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-09-28' and '2021-10-22'),

basePOS9 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-11-07' and '2021-12-07'),

baseNEG9 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-11-07' and '2021-12-07'),

basePOS10 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2021-12-24' and '2022-01-24'),

baseNEG10 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2021-12-24' and '2022-01-24'),

basePOS11 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2022-05-03' and '2022-05-15'),

baseNEG11 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2022-05-03' and '2022-05-15'),

basePOS12 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'Y' and time between '2022-06-06' and '2022-06-20'),

baseNEG12 AS

( SELECT \* from (

SELECT \*, CASE WHEN AmountBTC>0 THEN 'Y' ELSE 'N' END as direction from all\_transactions ) q

where direction = 'N' and time between '2022-06-06' and '2022-06-20')

SELECT \* FROM (

SELECT rnk1+rnk2+rnk3+rnk4+rnk5+rnk6+rnk7+rnk8+rnk9+rnk10+rnk11+rnk12+rnk13+rnk14+rnk15+rnk16+rnk17+rnk18+rnk19+rnk20+rnk21+rnk22+rnk23+rnk24 as score\_yep ,\* FROM (

SELECT o.Wallet, COALESCE(rnk1, 0) as rnk1, COALESCE(rnk2,0) as rnk2 , COALESCE(rnk3,0) as rnk3, COALESCE(rnk4, 0) as rnk4, COALESCE(rnk5,0) as rnk5 , COALESCE(rnk6,0) as rnk6, COALESCE(rnk7, 0) as rnk7, COALESCE(rnk8,0) as rnk8 , COALESCE(rnk9,0) as rnk9, COALESCE(rnk10, 0) as rnk10, COALESCE(rnk11,0) as rnk11 , COALESCE(rnk12,0) as rnk12,

COALESCE(rnk13, 0) as rnk13, COALESCE(rnk14,0) as rnk14 , COALESCE(rnk15,0) as rnk15, COALESCE(rnk16, 0) as rnk16, COALESCE(rnk17,0) as rnk17 , COALESCE(rnk18,0) as rnk18, COALESCE(rnk19, 0) as rnk19, COALESCE(rnk20,0) as rnk20 , COALESCE(rnk21,0) as rnk21, COALESCE(rnk22, 0) as rnk22, COALESCE(rnk23,0) as rnk23 , COALESCE(rnk24,0) as rnk24 from

(SELECT DISTINCT wallet from all\_transactions) o

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk1 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS1

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS1 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P1 ON o.wallet = P1.wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk2 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG1

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG1 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N1 ON o.Wallet = N1.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk3 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS2

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS2 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P2 ON o.Wallet = P2.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk4 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG2

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG2 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N2 ON o.Wallet = N2.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk5 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS3

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS3 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P3 ON o.Wallet = P3.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk6 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG3

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG3 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N3 ON o.Wallet = N3.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk7 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS4

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS4 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P4 ON o.Wallet = P4.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk8 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG4

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG4 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N4 ON o.Wallet = N4.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk9 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS5

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS5 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P5 ON o.Wallet = P5.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk10 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG5

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG5 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N5 ON o.Wallet = N5.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk11 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS6

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS6 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P6 ON o.Wallet = P6.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk12 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG6

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG6 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N6 ON o.Wallet = N6.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk13 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS7

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS7 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P7 ON o.Wallet = P7.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk14 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG7

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG7 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N7 ON o.Wallet = N7.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk15 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS8

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS8 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P8 ON o.Wallet = P8.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk16 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG8

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG8 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N8 ON o.Wallet = N8.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk17 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS9

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS9 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P9 ON o.Wallet = P9.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk18 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG9

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG9 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N9 ON o.Wallet = N9.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk19 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS10

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS10 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P10 ON o.Wallet = P10.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk20 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG10

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG10 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N10 ON o.Wallet = N10.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk21 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS11

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS11 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P11 ON o.Wallet = P11.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk22 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG11

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG11 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N11 ON o.Wallet = N11.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1 DESC) as rnk23 from (

select wallet, min(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from basePOS12

group by wallet, price ) x

where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from basePOS12 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) P12 ON o.Wallet = P12.Wallet

FULL JOIN(

select wallet, RANK() OVER(order by minprice1) as rnk24 from (

select wallet, max(price) as minprice1 from (

select wallet, price, avg(balanceBTC) as avgBalance, sum(abs(AmountBTC)) as sumAmount from baseNEG12

group by wallet, price ) x

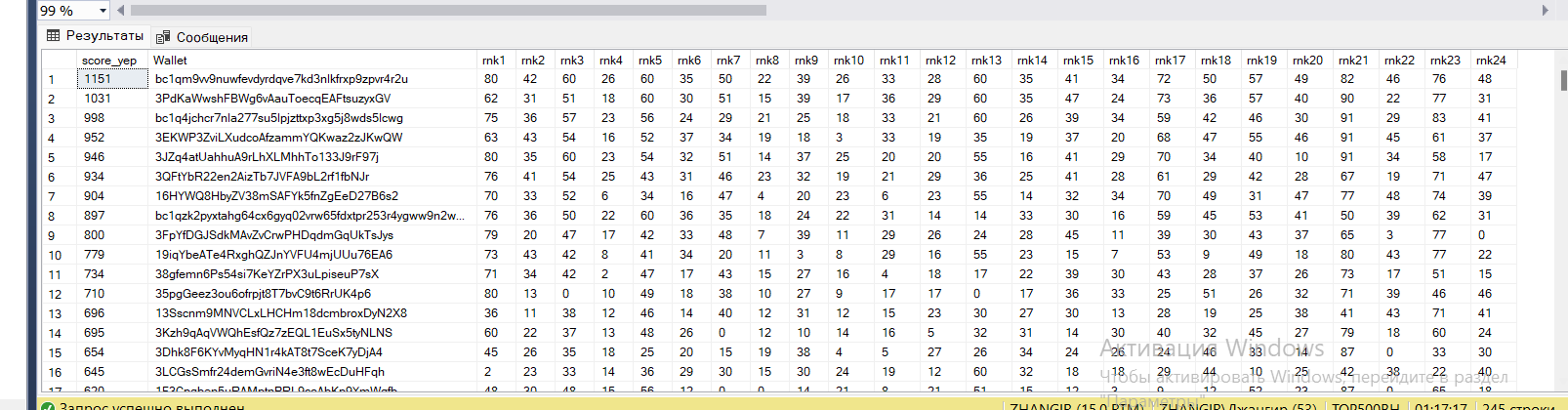
where sumAmount > 0.05\*(SELECT sum(abs(AmountBTC)) from baseNEG12 y where x.wallet = y.wallet ) or sumAmount> 0.01\*avgBalance

group by wallet ) q ) N12 ON o.Wallet = N12.Wallet ) isthisonlyone

) isthisonlyone2

WHERE score\_yep>0

order by score\_yep desc



bc1qm9vv9nuwfevdyrdqve7kd3nlkfrxp9zpvr4r2u

3PdKaWwshFBWg6vAauToecqEAFtsuzyxGV

bc1q4jchcr7nla277su5lpjzttxp3xg5j8wds5lcwg

3EKWP3ZviLXudcoAfzammYQKwaz2zJKwQW

3JZq4atUahhuA9rLhXLMhhTo133J9rF97j

3QFtYbR22en2AizTb7JVFA9bL2rf1fbNJr

16HYWQ8HbyZV38mSAFYk5fnZgEeD27B6s2

bc1qzk2pyxtahg64cx6gyq02vrw65fdxtpr253r4ygww9n2w2zjyy8wsnyc666

3FpYfDGJSdkMAvZvCrwPHDqdmGqUkTsJys

19iqYbeATe4RxghQZJnYVFU4mjUUu76EA6

38gfemn6Ps54si7KeYZrPX3uLpiseuP7sX

35pgGeez3ou6ofrpjt8T7bvC9t6RrUK4p6

13Sscnm9MNVCLxLHCHm18dcmbroxDyN2X8

3Kzh9qAqVWQhEsfQz7zEQL1EuSx5tyNLNS

3Dhk8F6KYvMyqHN1r4kAT8t7SceK7yDjA4

3LCGsSmfr24demGvriN4e3ft8wEcDuHFqh

1F3Cpgben5uRAMptnPRL9coAbKp9YmWqfb

bc1qfd9wg4eh4ey7ar9my04xk67mxf7rax8nkuwqjqwpttkcqj6wqkcq3kpx98

1P5ZEDWTKTFGxQjZphgWPQUpe554WKDfHQ

bc1qw8wrek2m7nlqldll66ajnwr9mh64syvkt67zlu

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_