**Введение**

Мой проект о крипторынке, а точнее о биткоине. Я анализирую все транзакции первых 500 держателей BTC (биткоина), на руках которых сконцетрировано около 30% всех биткоинов.

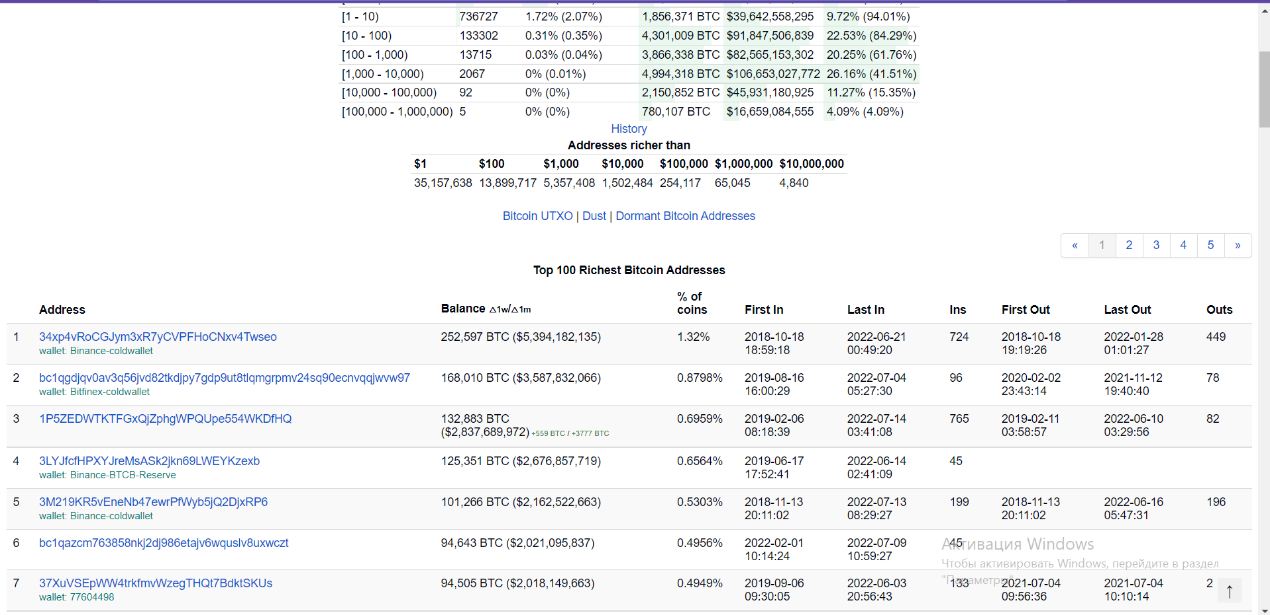
Особенность крипторынка в том, что он в сотню раз меньше, чем фондовый рынок, а потому серьезные люди, которые пришли на этот рынок с $миллиардами способны очень сильно менять цену и поведение всех криптоактивов, тем более если они будут координироваться с друг другом. Таких людей здесь называют "китами". Помимо них, существуют и "акулы", которые хоть владеют и меньшими активами, они более активны и порой показывают даже большие профиты ввиду большого количества выгодных сделок. Как правило они являются профессиональными трейдерами. Помимо технического и фундаментального анализа рынка, я считаю было хорошо взглянуть на то, что происходит наверху, среди больших игроков, так как именно они диктуют игру, искусно забирают все деньги у тех, кто не знает сдешних правил.

Еще хотел бы отметить, что наверное у всех на слуху, что криптовалюта - это полная анонимность транзакций. Однако все чуть по другому, да, ваш кошелек, через который вы проводите транзакции, невозможно привязать к какому то конкретному человеку, обеспечивая полную конфеденциальность, но сами транзакции с этими кошельками неизбежно сохраняются в интеренете навсегда. Именно благодаря этому у меня сейчас есть возможность начать этот проект.

( этот проект сделан только мной, все данные взяты из сайта ниже )

Использовать для начала я буду сайт <https://bitinfocharts.com/> , на котором постоянно обновляется список самых богатых держателей биткоина.

t<https://bitinfocharts.com/top-100-richest-bitcoin-addresses.html>



**Цели проекта**

Конечно же извлечь практическую выгоду. Я полагаю, что действия крупных игроков (киты) помогут лучше спрогнозировать долгосрочную ситуацию, в свою очередь "акулы" помогут понять краткосрочную ситуацию, так как среди них существуют люди, торующие в промежутках месяца, недели и даже дня (таких людей называют daytrader)

Глобальная цель проекта - изучить транзакции и поведение успешных кошельков, которые либо стабильно получают большой профит, либо угадывают направление рынка в большинстве случаев. Эта информация даст большое видение рынка на завтра и на год вперед.

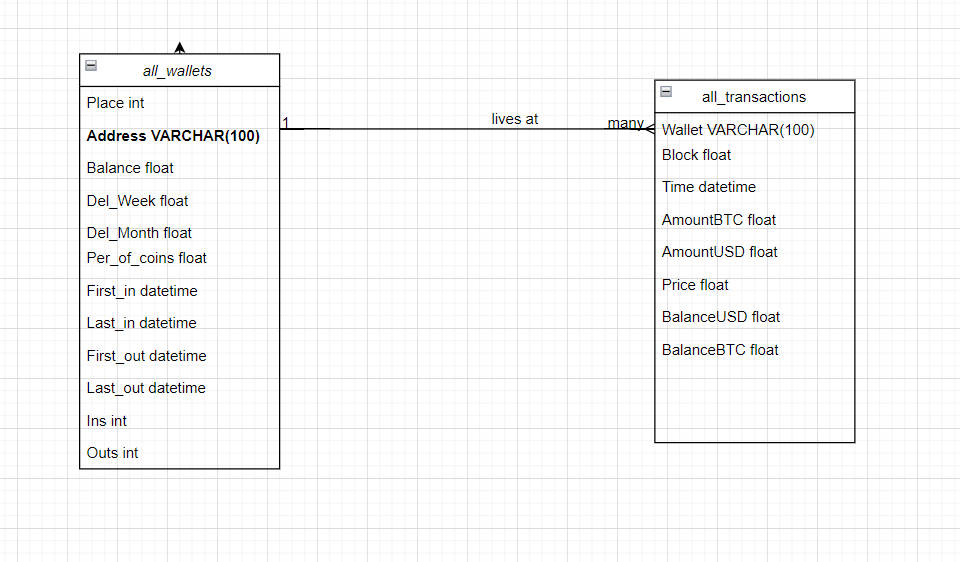
Вторичные цели, исходящие из первой:

* найти разного рода взаимосвязи между кошельками, 1) перебросы , 2) похожие транзакции
* составить список людей, которые часто прогнозируют цену в свою пользу
* составить список людей, чьи действия больше всего влияли на рынок в самых важных ситуациях при сильном падении или сильном росте.
* составить список людей (профессиональных трейдеров), проанализировать их транзакции и создать среднюю формулу, которая будет прогнозировать правильно по крайней мере в 50%+ случаев, желательно 60%+.

**Основная часть**

Большая часть анализа происходила на SQL ( MS SQL SERVER). Так как SQL заточен на работу с большими данными, запросы обычно не занимали больше 10 сек, что было очень важно в этом проекте, так как гипотез было много, запросов было много. Тем более, я владею этим инструментом на отличном уровне, поэтому я его считаю главным оружием в своем арсенале.

Самым первым этапом было получить необходимые данные из сайта, так как там не было готовой кнопки экспортировать файлы:) Для этого мне пришлось быстро изучить парсинг, (хочу заметить изучил я его на достаточно хорошем уровне). Я решил организовать все данные в две таблицы: all\_wallets (общая информация о кошельках) и all\_transactions (информация о транзакциях)



Парсинг происходил как на общей странице <https://bitinfocharts.com/top-100-richest-bitcoin-addresses.html>, меняя при этом цифры, так как таких страниц было 5, а также отдельно на странице каждого отдельного кошелька, так как там хранятся транзакции кошельков, к примеру <https://bitinfocharts.com/bitcoin/address/34xp4vRoCGJym3xR7yCVPFHoCNxv4Twseo-full>

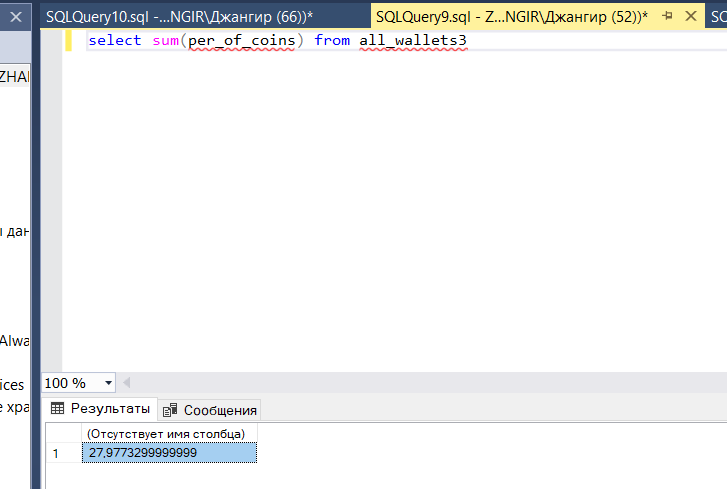
Все было проделано автоматически, и работает весь парсинг на трех файлах

***alltr.py , allwal.py , all\_a\_dict.json***

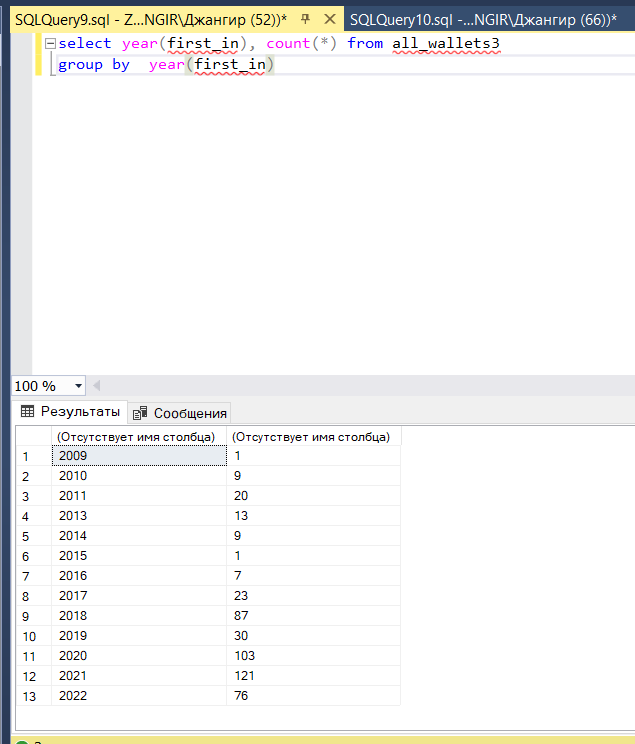
При загрузке в sql потребовалось записать данные в csv файлы

**all\_wallets.csv , all\_trans.csv**

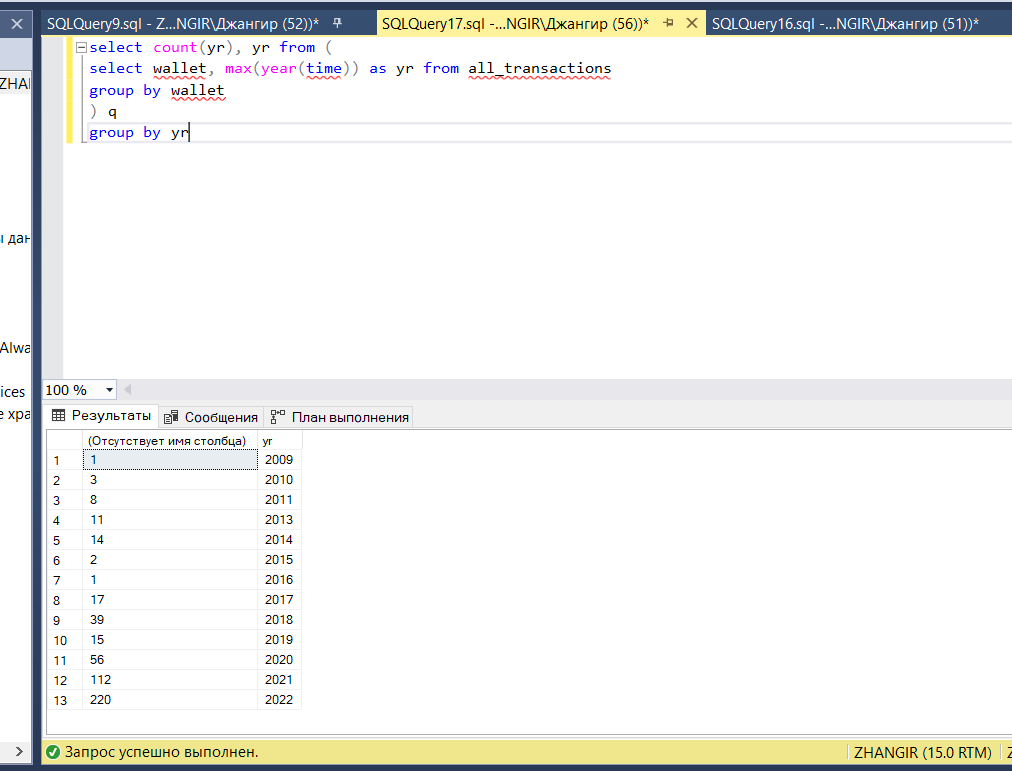
В первую очередь я проделал простенькие запросы для лучшего понимания того, с чем я работаю.



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



*Первое пополнение кошелька*



Последние операции с кошельков по годам

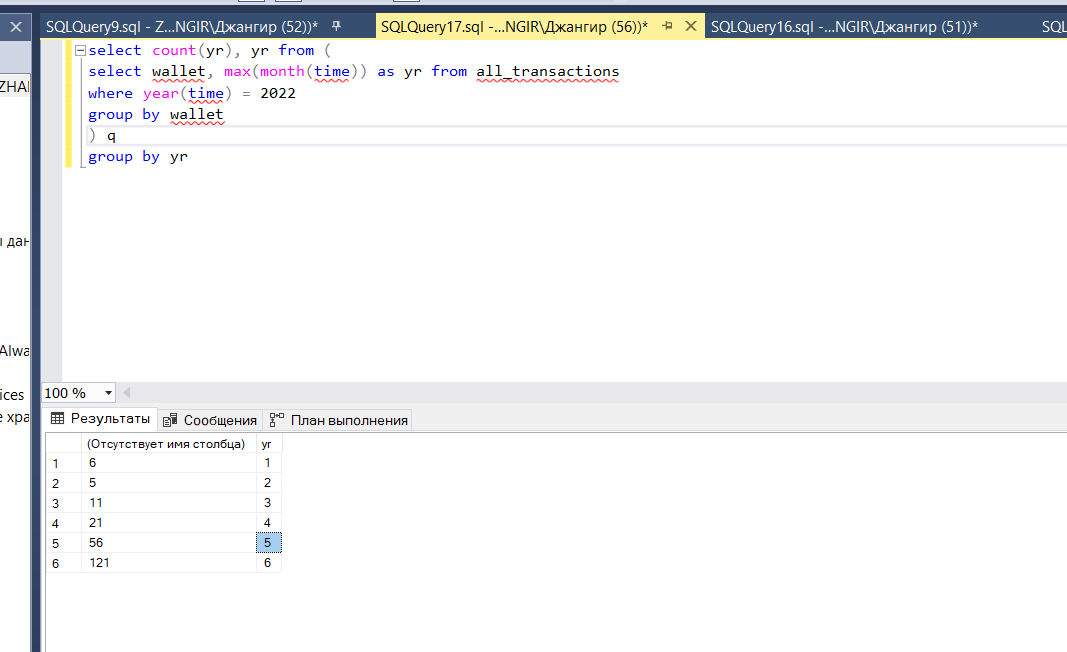
44% процента делали в этом году

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

Часть кошельков являются неактивными, часть из которых - забытые кошельки



Последние операции с кошельков по 2022 году (месяцам)

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

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

**СВЯЗАННЫЕ КОШЕЛЬКИ (ПЕРЕБРОСЫ)**

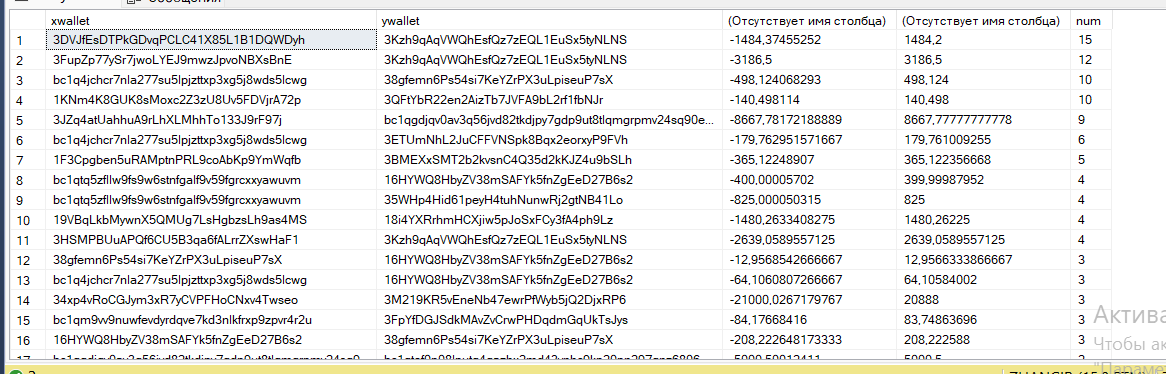
( убрал кошельки у которых очень много транзакций и которые составляют меньше 0.001 от их баланса )

( также поставил 0.99 разницу, так как считаю что комиссия не может быть больше 1%, либо это будут единичные случаи, которыми я готов 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 я буду брать в среднем 3 и более соответствий для точности результатов )

Применял статистические методы, чтобы избежать случайных совпадений



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

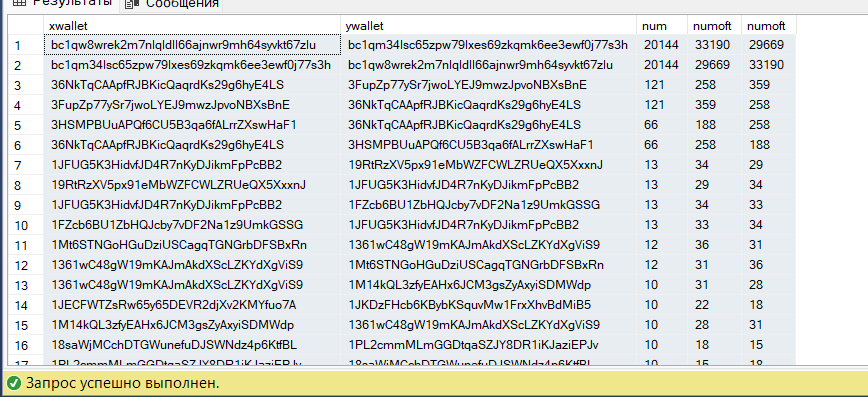
order by num desc

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

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

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

**СВЯЗАННЫЕ КОШЕЛЬКИ (ПОХОЖИЕ ТРАНЗАКЦИИ)**

****

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

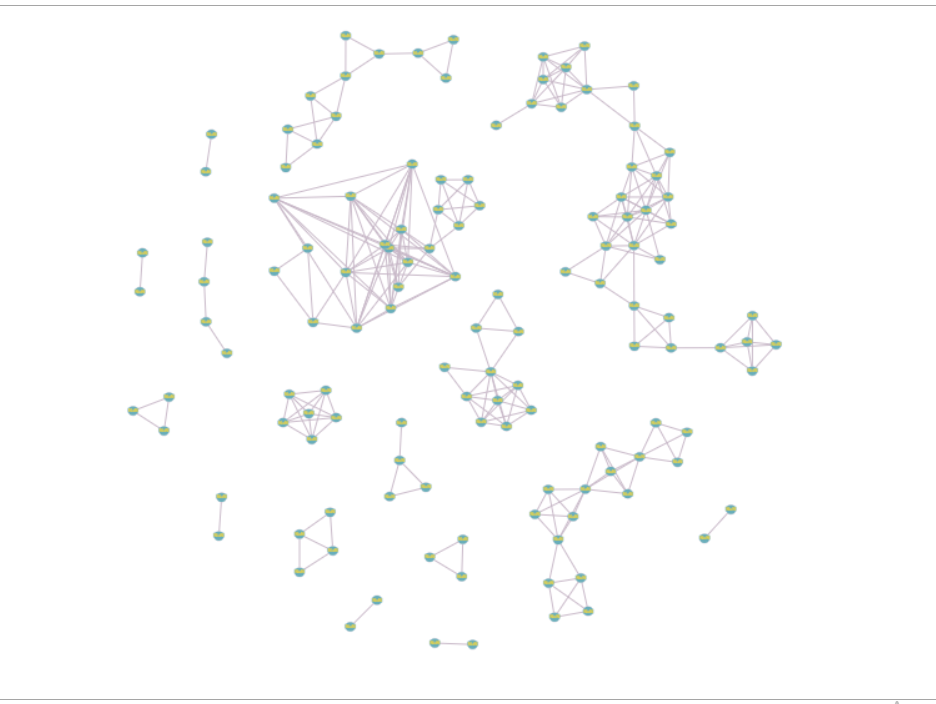
group by wallet  ) y on y.wallet = m.ywallet

where  num>2 and ((x.numoft+y.numoft)-2\*num)/coalesce(num, 1)<=5

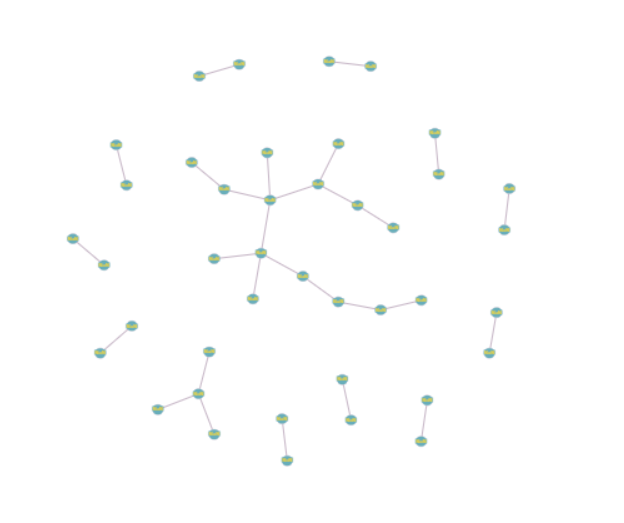
order by num desc

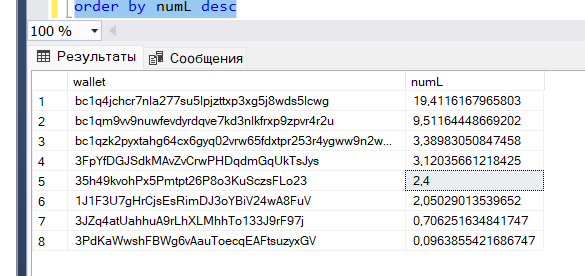
**ГРАФЫ ( СХЕМА ВЗАИМОСВЯЗЕЙ МЕЖДУ АККАУНТАМИ )**

**1)** СВЯЗАННЫЕ КОШЕЛЬКИ (ПЕРЕБРОСЫ)



2) СВЯЗАННЫЕ КОШЕЛЬКИ (ПОХОЖИЕ ТРАНЗАКЦИИ)

****



интересные кошельки (СКОРЕЕ ВСЕГО ТРЕЙДЕРЫ)

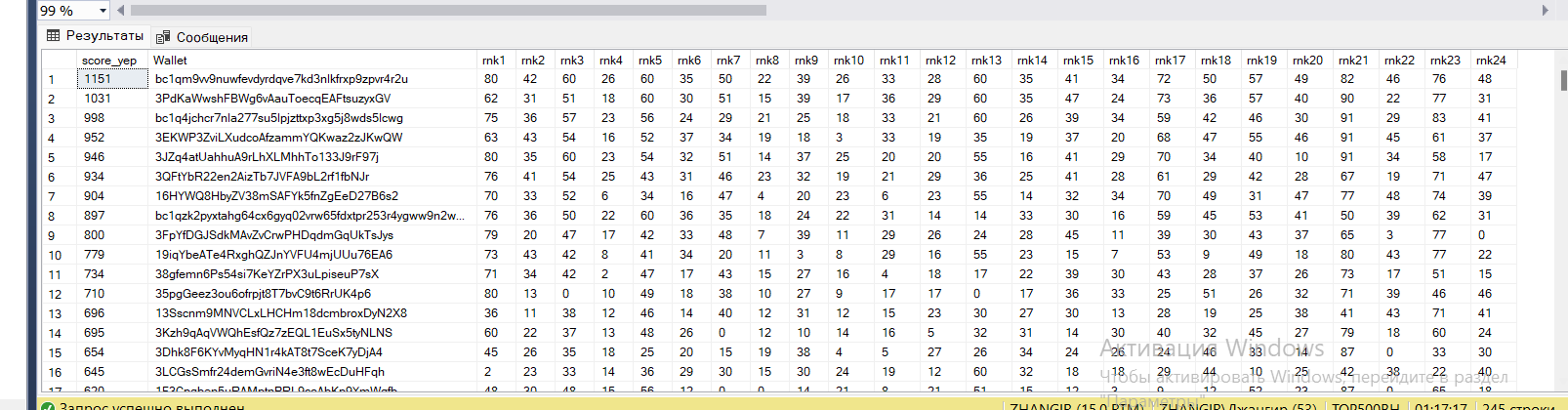
Graphical user interface, text, application

Description automatically generated

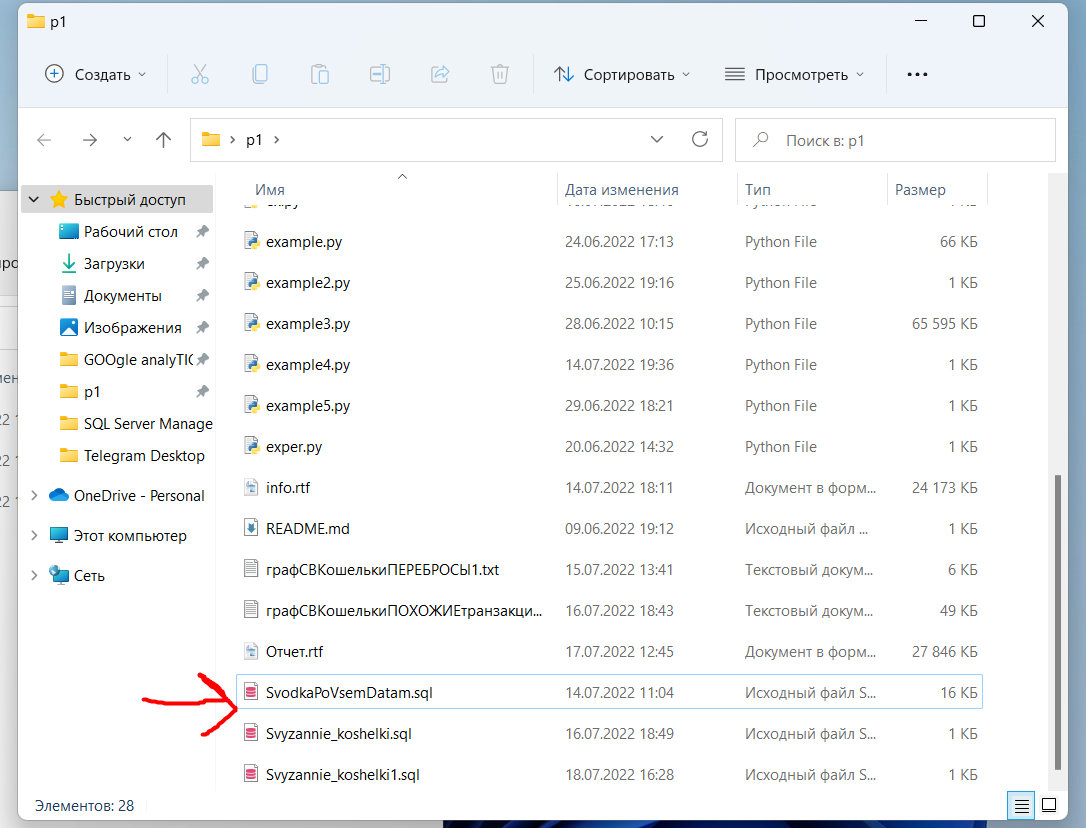
кошельки которые мы не будем брать в расчет в некоторых расчетах, по причине того что у них много транзакций на незначительную сумму, что иногда искажает данные. Хотя в некоторых расчетах мы их учитываем, убирая как раз эти транзакции или же объединяя их, допустим по дню, так как эти кошельки возможно как раз таки и представляют большую ценность.

**СОЗДАЛ ОЧЕНЬ КРУТОЙ СПОСОБ ОТСЛЕЖИВАТЬ ИНТЕРЕСНЫЕ КОШЕЛЬКИ ( идеального способа нет, но этот по крайней мере близок к этому )**

**Рейтинг людей ( по принципу того как часто они покупали по минимальной цене и продавали по максимальной цене в периоды сильного падения и сильного роста)**



**Так как код слишком большой, можете открыть файл SvodkaPoVsemDatam.sql**

****

Так как я считаю, что этот проект имеет какую-то ценность, и возможно не только для меня, я отправляю проект незаконченным, но думаю, что вы сможете оценить по тем навыкам, которые я применял здесь.

Даже у меня нет сейчас законченного проекта:) Я планирую дорабатывать его, добавить визуализации, так как данные обновляются, я планирую сделать автоматический сбор и анализ данных через каждый определенный период времени.

**Надеюсь проект был интересен**

**Хочу заметить, что мои навыки не ограничиваются теми, применяемые в проекте, к**

**тому же если что-то понадобится, я быстро учусь новому.**

**В своей работе также использовал сайт** [**https://graphonline.ru/**](https://graphonline.ru/)