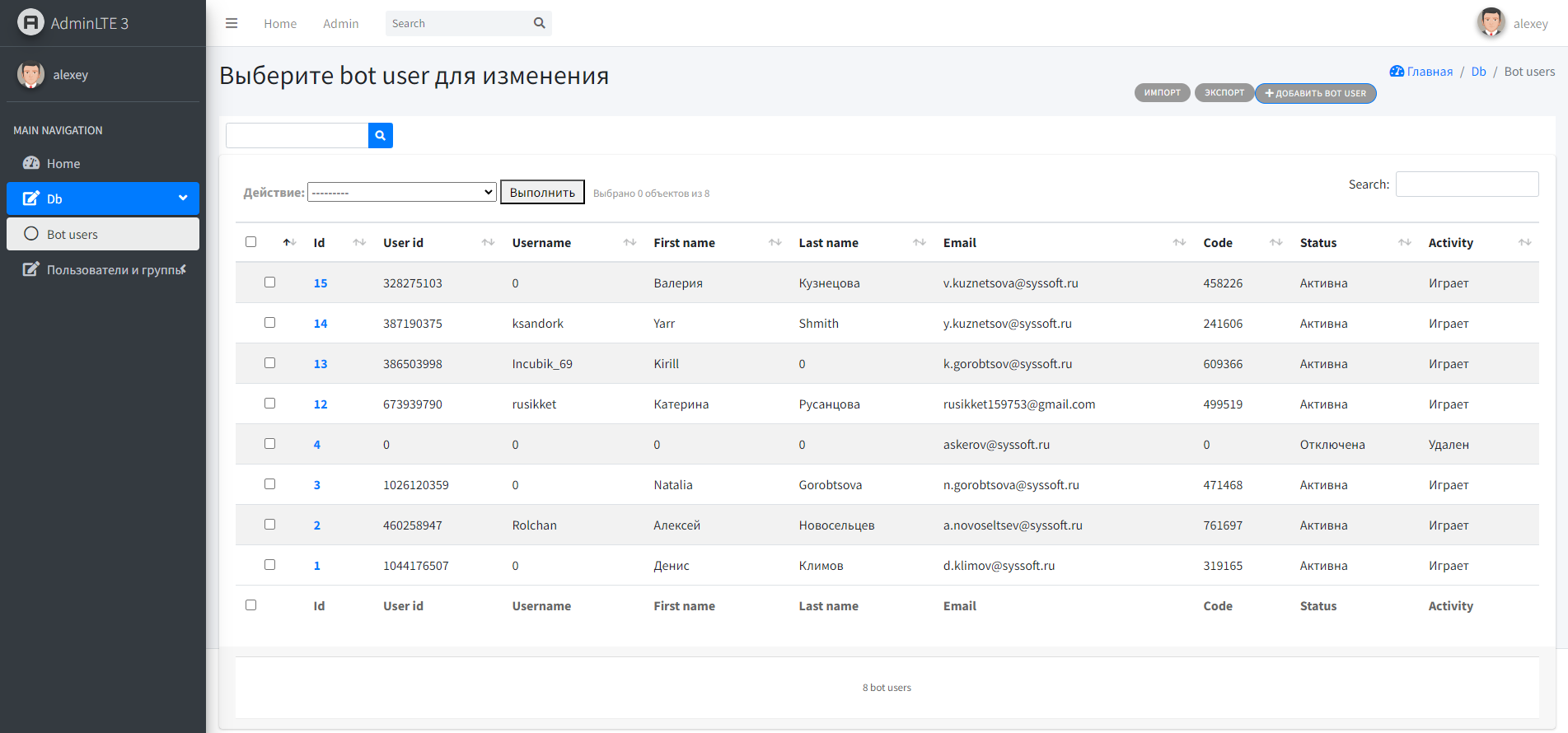
**syssoft\_random\_coffee\_bot**

После того как мы запускаем бота (python main.py), мы запускаем админку на Django (python admin\manage.py runserver).

Командой (python admin\manage.py createsuperuser) мы создаем пользователя для входа в админку.

<http://127.0.0.1:8000/admin/> - заходим в админку с данными пользователя, созданного командой (python admin\manage.py createsuperuser)

Переходим в Db - Bot users (здесь хранятся данные пользователей TG бота)



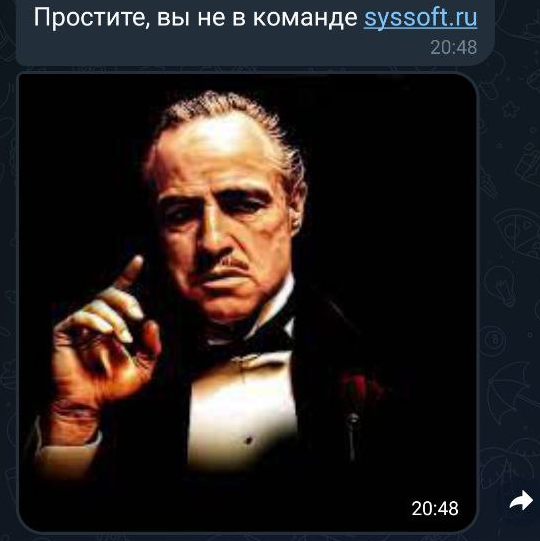
Через кнопку импорт мы должны импортировать в базу список e-mail адресов (excel-файлом), через которые пользователи будут регистрироваться в боте.

***Рекомендуется заполнять поле email в нижнем регистре!***



***Обязательно! Поля, которые не содержат данные – заполняются нулями!***

В боте есть проверка, есть ли почта в списке. Если нет, то бот выдаст сообщение:



После того, как электронные адреса будущих пользователей бота залиты в БД, пользователи могут приступить к регистрации в боте - <https://t.me/syssoft_random_coffee_bot>

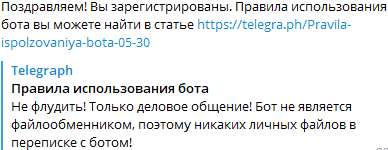
Когда пользователь нажимает кнопку /start, бот проверяет его User ID из TG на наличие в БД:

* если User ID уже содержится в БД (т.е. пользователь зарегистрирован) – "Вы уже зарегистрированы. Ждите понедельника!"
* если User ID отсутствует в БД – "Привет, {first\_name из TG}! Укажите свой адрес электронной почты."

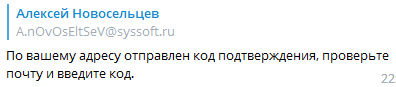
После ответа пользователя, бот проверяет сообщение на наличие в нем почты, затем проверяет **домен** на наличие в списке разрешенных

***\*список изменяется в файле config.py в строке DOMAINS***

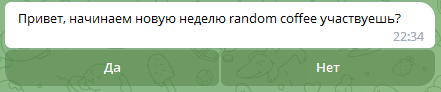
Если почта в списке БД, то начинается процесс отправки кода подтверждения на указанную почту и ожидание ввода кода пользователем. Если этап пройден успешно – пользователь регистрируется в боте со статусом «Играет»



***\*бот превращает сообщение почтой в нижний регистр, чтобы избежать ошибок***

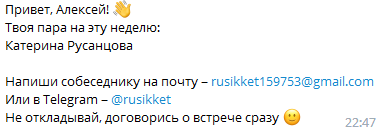
******

В понедельник в 9:00 всем пользователям со статусом почты "Активна" приходит сообщение от бота:



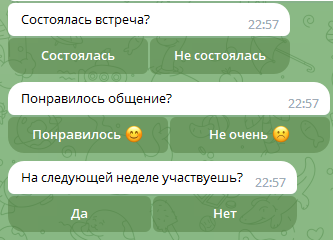
* если нажать "**Да**", то бот сообщит "Вы приняли участие в random coffee!" и статус в БД останется "Играет"
* если нажать "**Нет**", то бот сообщит "Вы отказались от участия в random coffee!" и статус в БД изменится на "Пауза"

В понедельник в 9:15 всем пользователям со статусом почты "Активна" и активностью "Играет" приходит сообщение от бота:



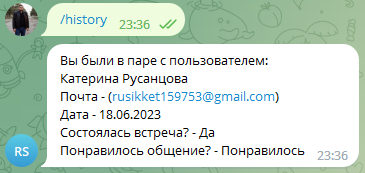
* имя автоматически меняется для каждого пользователя
* если в TG не заполнены (имя, фамилия, почта или ник), то в сообщении они заменятся соответственно (имя отсутствует, фамилия отсутствует, почта отсутствует или ник отсутствует)
* поэтому рекомендую использовать исключительно внутреннюю почту «syssoft.ru», чтобы можно было без проблем найти пользователя в TrueConf и связаться с ним

В пятницу в 17:00 всем пользователям со статусом почты "Активна" и активностью "Играет" приходит сообщение от бота:



Первые два ответа сохраняются в истории пары.

Через меню можно вызвать команду "/history", которая выведет все ваши пары:



Последний сохраняет или меняет активность по принципу:

* если нажать "**Да**", то бот сообщит "Вы приняли участие в random coffee!" и статус в БД останется "Играет"
* если нажать "**Нет**", то бот сообщит "Вы отказались от участия в random coffee!" и статус в БД изменится на "Пауза"

За все рассылки отвечает **scheduler**, **day\_of\_week** – первые три буквы для недели по-английски, **hour** – время от 0 до 24, а **minute** – минуты от 0 до 60.

if \_\_name\_\_ == '\_\_main\_\_':  
 # Запускаем функцию send\_coffee\_pairs() при запуске бота  
 scheduler = AsyncIOScheduler(event\_loop=bot.loop)  
 scheduler.add\_job(send\_game\_question, 'cron', day\_of\_week='mon', hour=9, minute=0)  
 scheduler.add\_job(send\_coffee\_pairs, 'cron', day\_of\_week='mon', hour=9, minute=15)  
 scheduler.add\_job(send\_survey, 'cron', day\_of\_week='fri', hour=17, minute=0)  
 scheduler.start()  
  
 # Запускаем бота  
 executor.start\_polling(dp, skip\_updates=True)

Все логи записываются в файл **app.log** в корне проекта.

**2023-06-18 23:46:30,537** - **INFO** - **Start polling**.

Состоят из трех частей:

* дата и время
* типы сообщений: DEBUG, INFO, WARNING, ERROR или CRITICAL
* и само сообщение