

Алгоритмический бизнес

Telegram бот для общения с клиентами своими
руками

Где наши клиенты?

Мессенджеры позволяют общаться с клиентом наиболее естественным образом

Сделаем своего bota в Telegram

Amazon создает свой мессенджер



Тестируем Диалоги в Сбербанк Онлайн (iOS)



Примеры общения с клиентами

Сбербанк в Telegram



1. СбербанкБот_dev
2. Вкус-вилл
3. Мой МТС: бот-ассистент
4. Talk bank
5. KLM Facebook Messenger bot

в Telegram можно встретить виртуальный банк

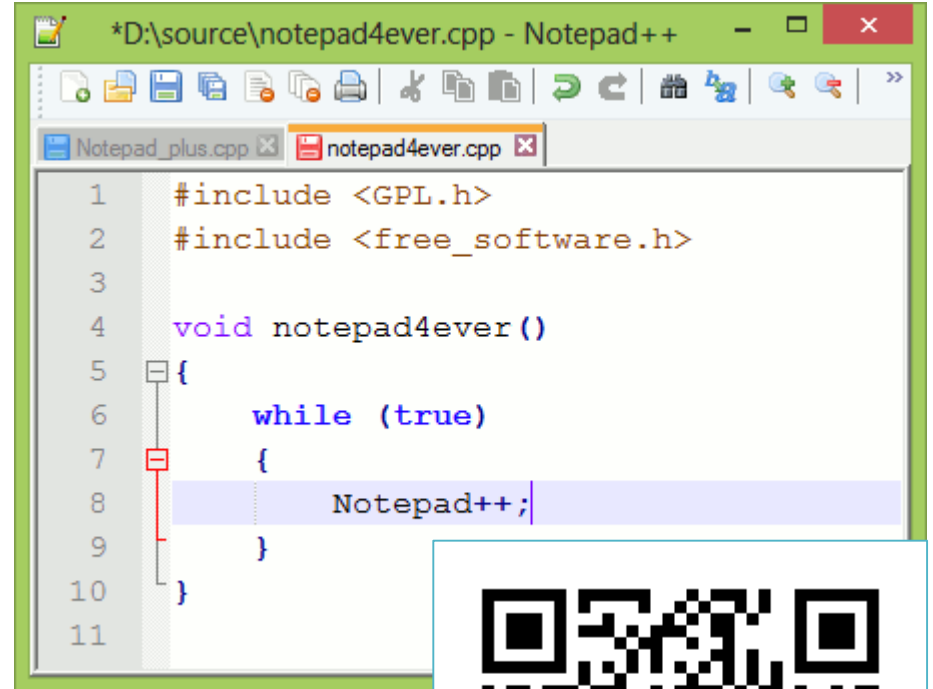


Python - это язык решения задач



1. Интеграция компонентов
(*Uber, Spotify, Yandex, Telegram и другие*)
2. Быстрое создание прототипов
3. Распространяется бесплатно
4. Один из самых простых в изучении
5. Программирование математических вычислений, Machin Learning
6. Python — активно развивающийся язык программирования
7. Научиться говорить с компьютером

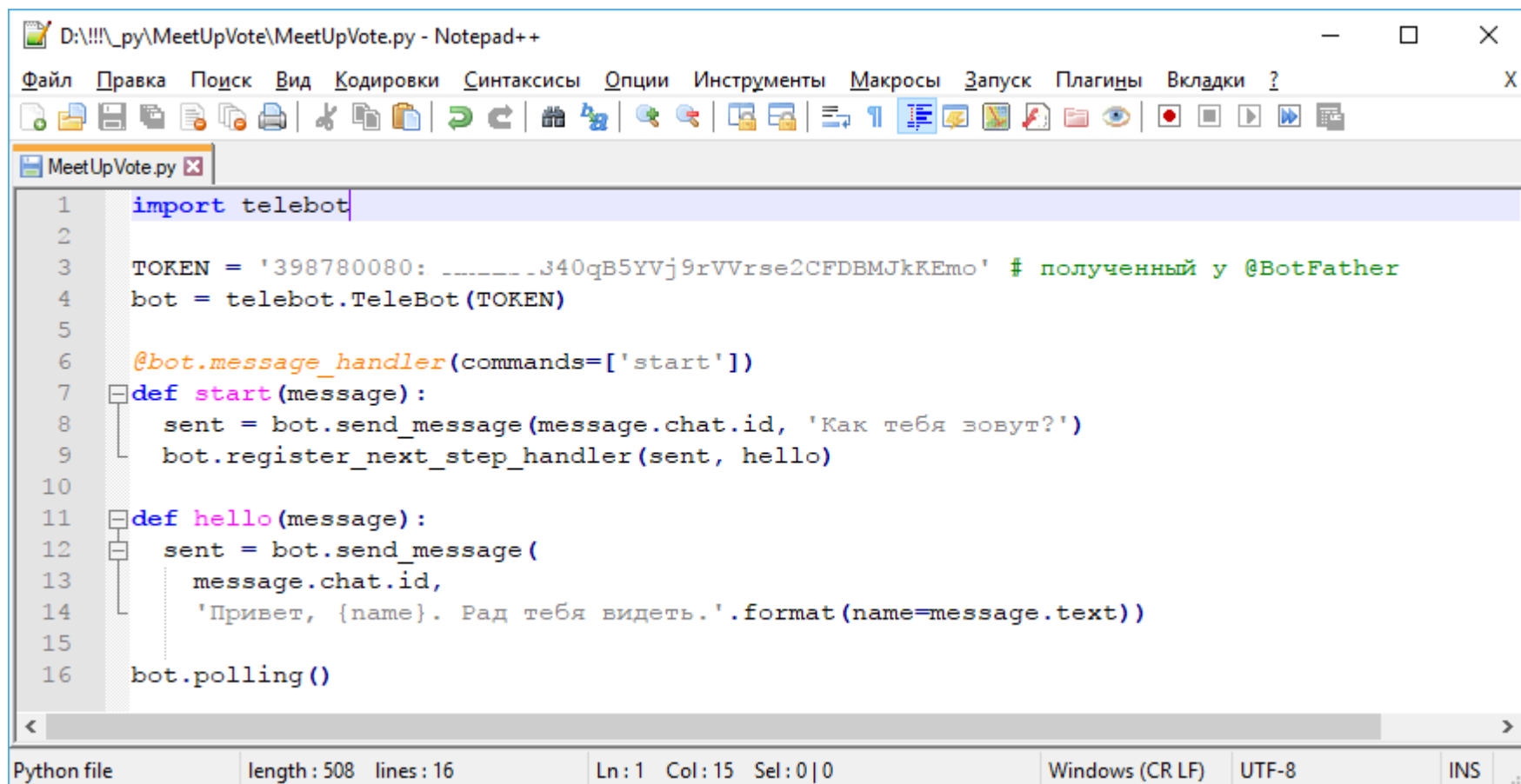
Потребуется инструменты

A screenshot of a Notepad++ window titled "*D:\source\notepad4ever.cpp - Notepad++". The window shows a C++ program with the following code:

```
1  #include <GPL.h>
2  #include <free_software.h>
3
4  void notepad4ever ()
5  {
6      while (true)
7      {
8          Notepad++;
9      }
10 }
11
```

The code is syntax-highlighted, and the line "Notepad++;" is highlighted in blue. The window has a standard toolbar and a tab bar showing "Notepad_plus.cpp" and "notepad4ever.cpp".

Создаем модуль с кодом



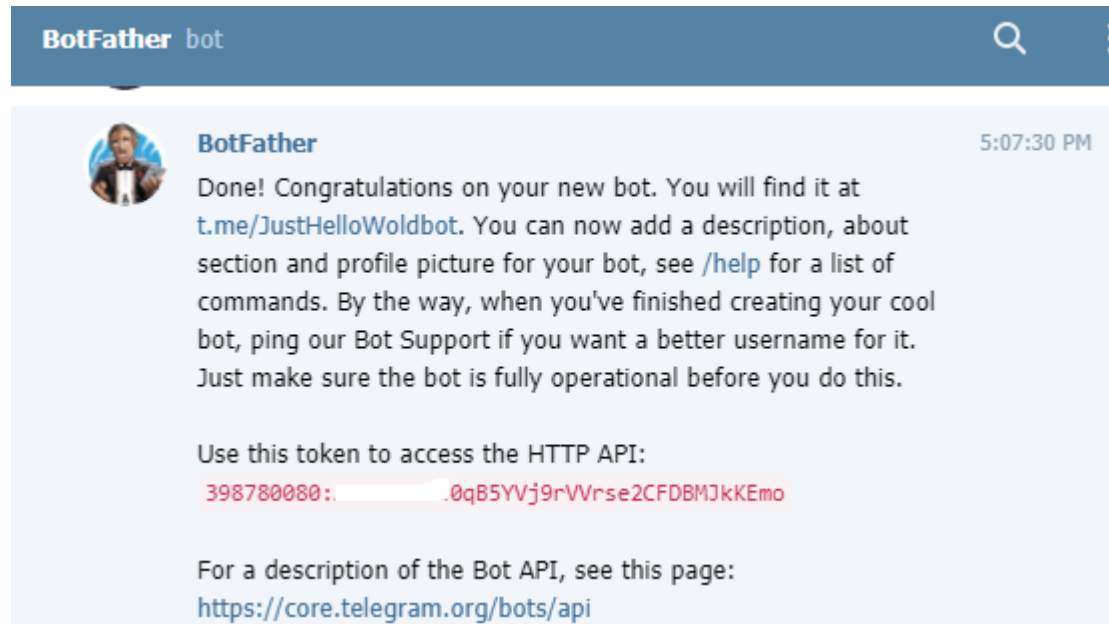
The image shows a Notepad++ window titled "D:\!!!_py\MeetUpVote\MeetUpVote.py - Notepad++". The menu bar includes "Файл", "Правка", "Поиск", "Вид", "Кодировки", "Синтаксисы", "Опции", "Инструменты", "Макросы", "Запуск", "Плагины", "Вкладки", and "?". The toolbar contains various icons for file operations, editing, and running. The active tab is "MeetUpVote.py". The code is as follows:

```
1 import telebot
2
3 TOKEN = '398780080: _____J40qB5YVj9rVVrse2CFDBMJkKEmo' # полученный у @BotFather
4 bot = telebot.TeleBot(TOKEN)
5
6 @bot.message_handler(commands=['start'])
7 def start(message):
8     sent = bot.send_message(message.chat.id, 'Как тебя зовут?')
9     bot.register_next_step_handler(sent, hello)
10
11 def hello(message):
12     sent = bot.send_message(
13         message.chat.id,
14         'Привет, {name}. Рад тебя видеть.'.format(name=message.text))
15
16 bot.polling()
```

The status bar at the bottom shows "Python file", "length: 508 lines: 16", "Ln: 1 Col: 15 Sel: 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".

Регистрируем своего бота

1. Связываемся с BotFather
2. Создаем своего бота по команде /newbot
3. Получаем ключевой текст – token:



BotFather



Придумаем сценарий обслуживания клиентов

1. Создание Выступающего
2. Регистрация Слушателей
3. Переписка Выступающий -> всем Слушателям
4. Любой Слушатель -> только Выступающему
5. Потребуется базой данных

Meet Up Vote bot



Давайте пообщаемся:

Листинг на 1 страницу

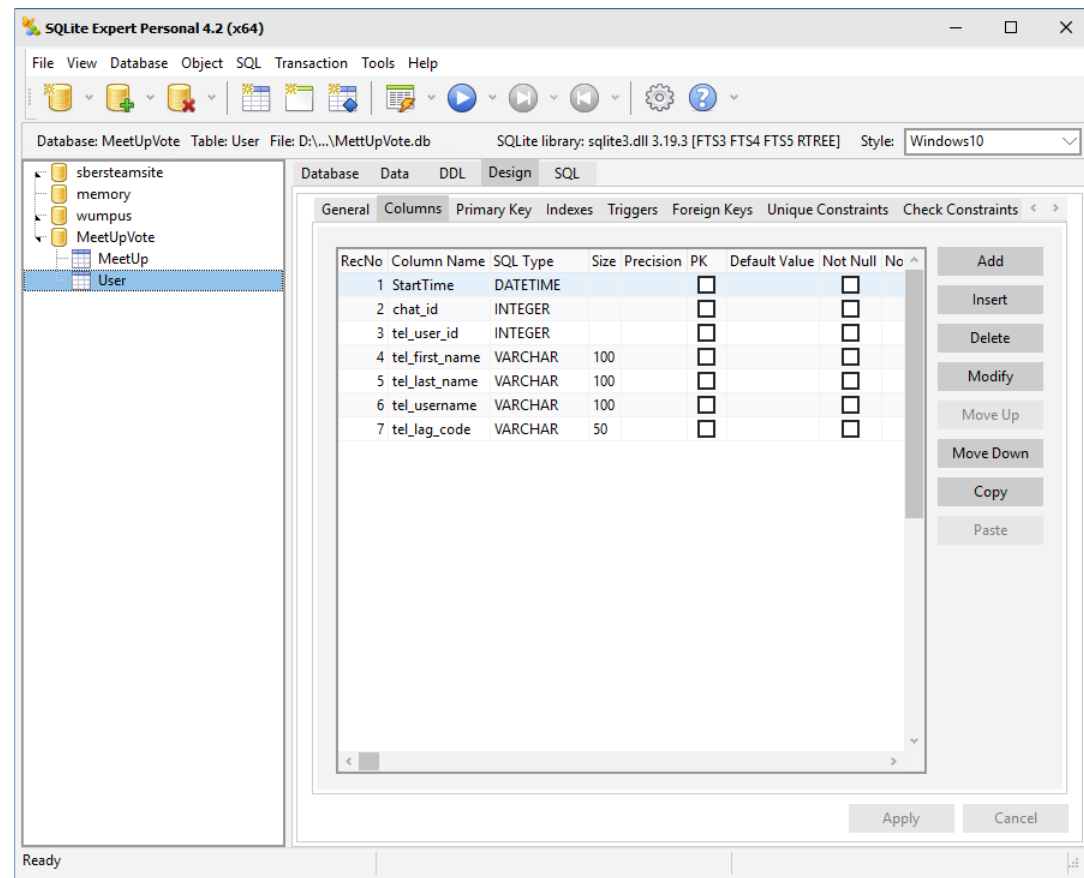
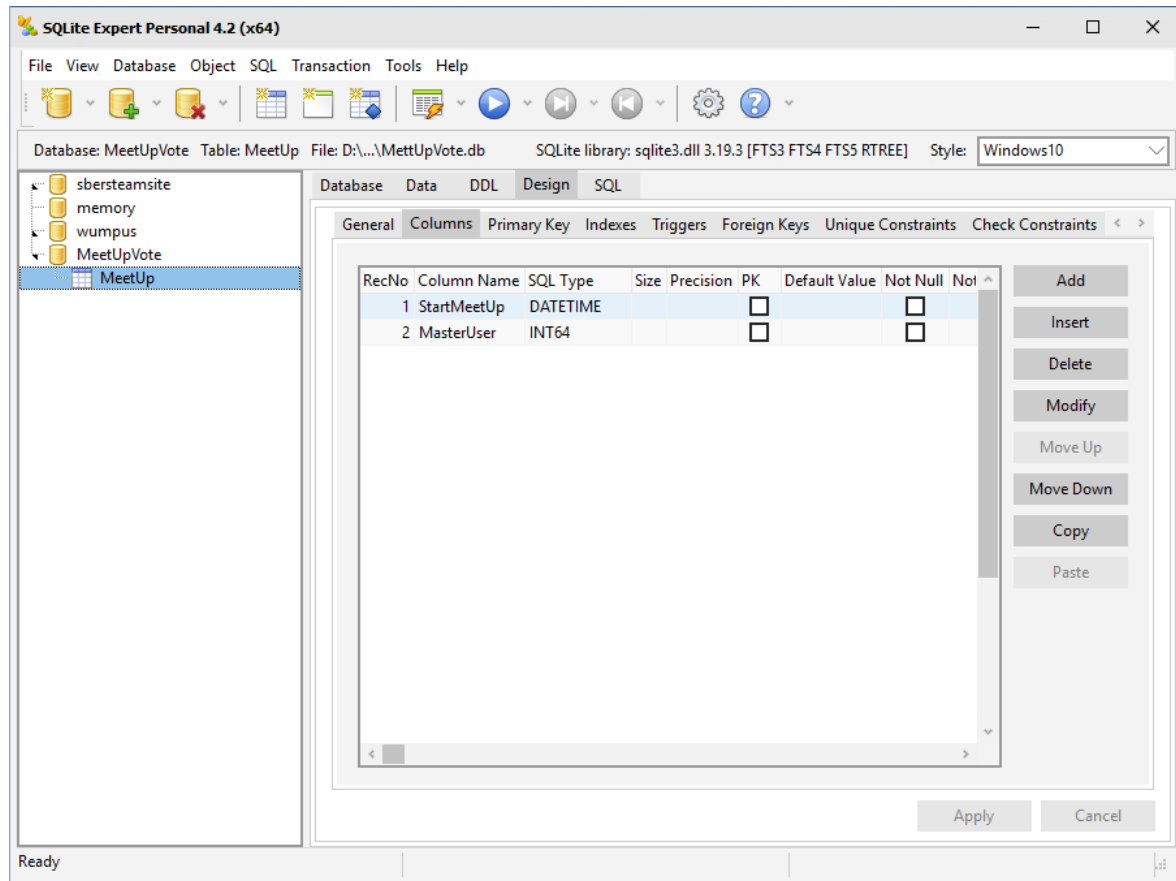
```
*D:\!!!\py\MeetUpVote\MeetUpVote2.py - Notepad++
Файл Правка Поиск Вид Кодировки Синтаксисы Опции Инструменты Макросы Запуск Плагины Вкладки ?
MeetUpVote2.py x

1 import telebot
2 import sqlite3
3 class SQLighter:
4     def __init__(self, database_name):
5         self.database_name = database_name
6         self.cursor = None
7         self.connection = None
8     def connect(self):
9         self.connection = sqlite3.connect(self.database_name)
10        self.cursor = self.connection.cursor()
11    def execute(self, sql):
12        self.cursor.execute(sql)
13        self.connection.commit()
14    def fetchall(self):
15        return self.cursor.fetchall()
16    def close(self):
17        self.cursor.close()
18        self.connection.close()
19
20 TOKEN = '339578780:RzLhus16fGm25XjPd1l8bw70do' # получаемый у @BotFather
21 database_name = 'MeetUpVote.db'
22 MEETUP_TIMEOUT = '7200'
23 bot = telebot.TeleBot(TOKEN)
24 @bot.message_handler(commands=['start'])
25 def start(message):
26     db_worker = SQLighter(database_name)
27     db_worker.cursor.execute("INSERT into User (StartTime,chat_id,tel_user_id,tel_first_name,tel_last_name,tel_username,tel_lang_code,nic) VALUES(CURRENT_TIMESTAMP,(?),(?),(?),(?),(?),(?),(?))",
28                             (message.chat_id, message.from_user.id, message.from_user.first_name, message.from_user.last_name, message.from_user.username, message.from_user.language_code, message.text)).fetchall()
29     db_worker.connection.commit()
30     Meet = db_worker.cursor.execute("SELECT * FROM MeetUp WHERE ((strftime('%s','now') - strftime('%s',StartTime)) < [time]) order by StartTime desc".format(time=MEETUP_TIMEOUT)).fetchall()
31     if Meet:
32         db_worker.close()
33         @bot.message_handler(commands=['master'])
34         def master(message):
35             db_worker = SQLighter(database_name)
36             db_worker.cursor.execute("INSERT into MeetUp (StartTime, MasterUser, MeetUpName) VALUES(CURRENT_TIMESTAMP,(?),(?))",
37                                     (message.chat_id, message.text)).fetchall()
38             if db_worker.cursor.lastrowid:
39                 bot.send_message(message.chat_id, 'Что-то пошло не так. Повтори создание встречи')
40                 db_worker.connection.commit()
41                 db_worker.close()
42             @bot.message_handler(func=lambda m: True)
43             def echo_all(message):
44                 db_worker = SQLighter(database_name)
45                 Meet = db_worker.cursor.execute("SELECT rowid,* FROM MeetUp WHERE ((strftime('%s','now') - strftime('%s',StartTime)) < [time]) order by StartTime desc".format(time=MEETUP_TIMEOUT)).fetchall()
46                 if Meet:
47                     Users = db_worker.cursor.execute("SELECT u.* FROM User u, MeetUp m WHERE (u.StartTime > m.StartTime) and (m.rowid = [mid])".format(mid=Meet[0][0])).fetchall()
48                     if Users:
49                         for u in Users:
50                             if (u[1] != message.chat_id) and (message.chat_id == Meet[0][2]):
51                                 bot.send_message(u[1], Meet[0][3] + ": " + message.text)
52                             elif (message.chat_id != Meet[0][2]) and (u[1] == message.chat_id):
53                                 bot.send_message(Meet[0][2], u[7] + ": " + message.text)
54                     else:
55                         bot.send_message(message.chat_id, 'Нет зарегистрированных посетителей')
56                 else:
57                     bot.send_message(message.chat_id, 'Просьба зарегистрировать на встрече командой /start')
58                 db_worker.close()
59
60 bot.polling()
```

Python file length: 3 478 lines: 60 Ln: 60 Col: 3 Sel: 0|0 Windows (CR LF) UTF-8 INS

workshop

Создаем базу



Вопросы к аудитории? Обсуждаем, голосуем в Telegram

1. Давайте подумаем какие финансовые кейсы могут возникнуть?
2. Что потребуются от финансовой платформы? Как определять получателя и отправителя средств? Какие данные?
3. Как получить номер телефона? Как инициировать p2p?

Meet Up Vote bot



Знаешь что такое AI NLP?

ищем таланты

Об авторе

Шашков Владислав

Product owner

Data-driven products

Дивизион «Цифровая платформа»

+7-985-660-55-67



Скоротать время затейливо играя в Telegram

Ощутить романтику самых первых игр на компьютерах позволит винтажная игра «Охота на Монстра» в Telegram

Поделись впечатлением в конце каждой игры 😊



Hunt The Wumpus bot

