**Main.py**

*import* telebot  
*import* requests  
*from* bs4 *import* BeautifulSoup  
*from* telebot *import* types  
*from* geopy *import* Yandex  
*from* geopy.distance *import* geodesic  
*import* pandas *as* pd  
*from* pandas *import* read\_csv  
   
  
bot = telebot.TeleBot("5336357947:AAGjQ7Sxii\_RrCN2aS0HUKxOxADM6Jy5miY")  
  
  
@bot.message\_handler(commands=['start'])  
*def* start(*message*):  
 markup = types.ReplyKeyboardMarkup(resize\_keyboard= *True*)  
 item1 = types.KeyboardButton('Найти лекарство по названию')  
 item2 = types.KeyboardButton('Найти ближайшую ко мне аптеку')  
  
 markup.add(item1, item2)  
  
 bot.send\_message(*message*.chat.id, 'Привет {0.first\_name}, я Таблеткин и я могу тебе помочь найти лекарства или ближайшую '  
 'аптеку, реши что тебе нужно и просто нажми кнопку?'.format(*message*.from\_user), reply\_markup=markup)  
  
  
@bot.message\_handler(content\_types=['text'])  
*def* find(*message*):  
 *if message*.text == 'Найти лекарство по названию':  
 bot.send\_message(*message*.chat.id, "Введите название, препарата который ищете")  
 bot.register\_next\_step\_handler(*message*, search)  
  
 *if message*.text == 'Найти ближайшую ко мне аптеку':  
 bot.send\_message(*message*.chat.id, "Введите ваш адрес пожалуйста, \nПример: 'Московская 267/1, Брест' ")  
 bot.register\_next\_step\_handler(*message*, find\_apteka)  
  
  
#ПОИСК ЛЕКАРСТВ--------------  
*def* search(*message*):  
 bot.send\_message(*message*.chat.id, "Начинаю поиск лекарства")  
  
 url = 'https://tabletka.by'  
  
 find\_url = 'https://tabletka.by/search?request=' + *message*.text  
 response = requests.get(find\_url)  
 soup = BeautifulSoup(response.text, 'lxml')  
  
 save\_message = *message*.text  
 capitalizeMessage = save\_message.capitalize()  
  
  
 internalLinks = [  
 a.get('href')  
 *for* a *in* soup.find\_all('a')  
 *if* a.get('href') *and* a.get('href').startswith('/result') *and* a.get\_text().startswith(f"{capitalizeMessage}")]  
  
  
 print(capitalizeMessage)  
 sorted\_links = []  
 *for* j *in* internalLinks:  
 *if* j *not in* sorted\_links:  
 sorted\_links.append(j)  
  
  
 *if* sorted\_links == []:  
 bot.send\_message(*message*.chat.id, "По вашему запросу ничего не найдено")  
  
 *for* i *in* range(len(sorted\_links)):  
 result = sorted\_links[i]  
 new\_url = url + result  
 bot.send\_message(*message*.chat.id, new\_url)  
 *if* i == 9:  
 *break*#ПОИСК АПТЕКИ----------------------  
*def* find\_apteka(*message*):  
 place = *message*.text  
 location = Yandex(api\_key='0ef27e34-3ede-402a-95f2-fee33edd3e6f').geocode(place)  
 print(location)  
  
 *try*:  
 lat = location.latitude  
 lon = location.longitude  
 *except* AttributeError:  
 print("Введённые данные некорректны")  
 *else*:  
 print("Всё ок")  
 latitude\_place = float(lat)  
 longitude\_place = float(lon)  
  
 path = "D:/6\_semester/CourseWork/apt\_info.csv"  
 df1 = read\_csv(path, delimiter=",", header=0)  
  
 apt\_coords = []  
 loc\_dist = {}  
 *for* i, j, k *in* zip(df1['Latitude'], df1['Longitude'], df1['Name']):  
 temp\_apteka = (i, j)  
 apt\_coords.append(temp\_apteka)  
 loc\_dist.update({f"{k}": f"{temp\_apteka}"})  
  
 user = (f"{latitude\_place}", f"{longitude\_place}")  
  
 distance\_m = []  
 *for* i *in* range(len(apt\_coords)):  
 distance\_m.append(geodesic(user, apt\_coords[i]).meters)  
  
 count = 0  
  
 *for* key *in* loc\_dist:  
 loc\_dist[key] = distance\_m[count]  
 count += 1  
  
  
 min\_dist = min(loc\_dist.values())  
  
 apteka\_name = list(loc\_dist.keys())[list(loc\_dist.values()).index(min\_dist)]  
 *for* i, j, k, z *in* zip(df1['Name'], df1['Address'], df1['Telephone'], df1['Work-time']):  
 *if* apteka\_name == i:  
 print(i, j, k, z)  
 bot.send\_message(*message*.chat.id,  
 f"{i}" + "\nАдрес: " + f"{j}" + "\nТелефон: " + f"{k}" + "\nВремя работы: " + f"{z}")  
  
  
bot.polling(none\_stop = *True*)