# **Руководства по работе с ПО**

## **Разработка руководства администратора**

Для начало работы, надо скачать файлы скриптов интерфейса и парсинга. [Ссылка](https://drive.google.com/drive/folders/1dvk51oxOXovkhyJ5nWEJHfZLB0scOVwu?usp=sharing)

Скачав, распакуем файл All\_modules и запускаем файл All\_modules.exe.

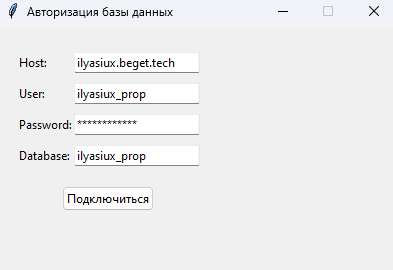


Рисунок 1 - окно авторизации

Сейчас нам открылось окно авторизации, в котором надо ввести параметры своей базы данных. После ввода, нажимаем на кнопку “Подключиться”. Если данные введены правильно и база существует то выйдет сообщение:

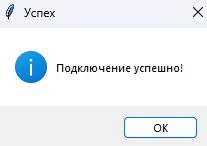


Рисунок 2 - успех

После успешного входа нам открывается интерфейс для настроек.

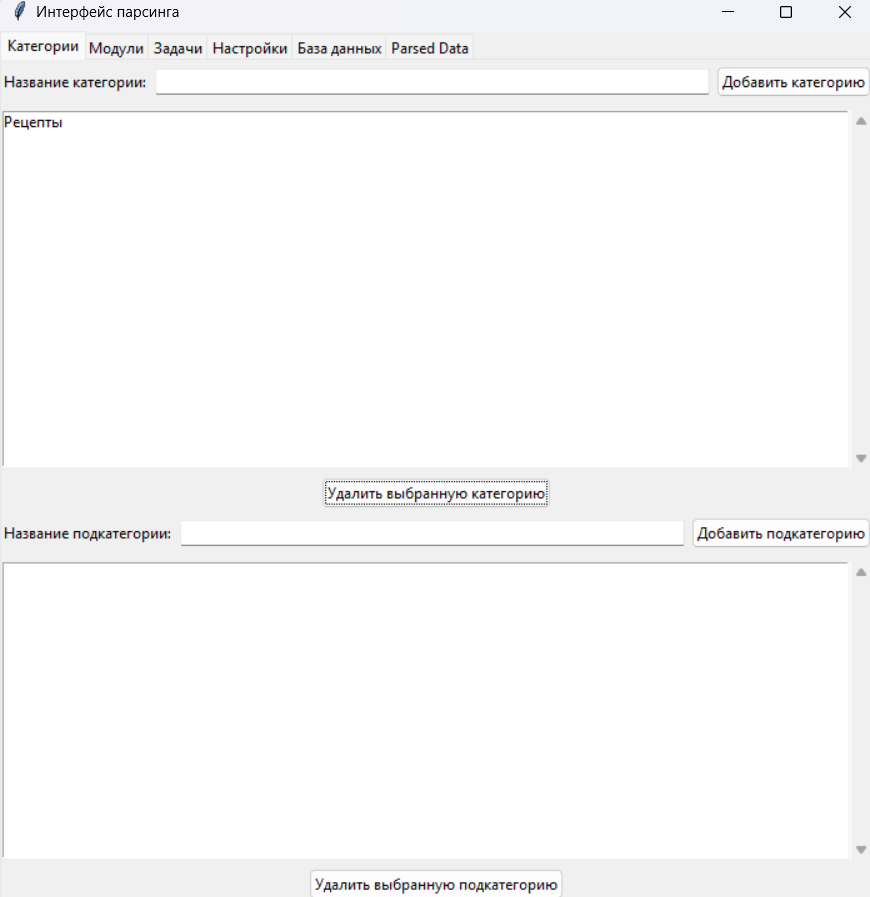


Рисунок 3 - интерфейс

Здесь у нас 6 вкладок: Категории, Модули, Задачи, Настройки, База данных, Parsed Data. Начнем с вкладки “Категории”. Здесь можно создать категорию и подкатегорию. Вначале создадим категорию “Тест1”.

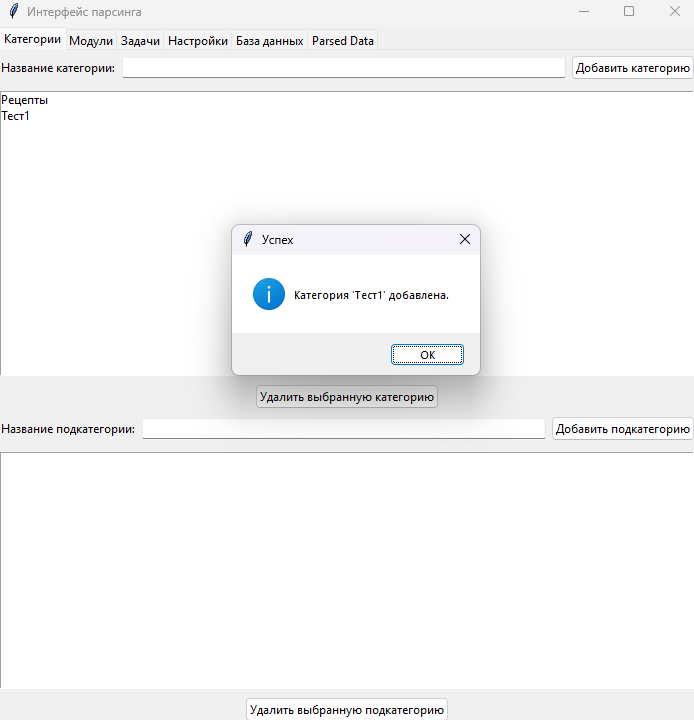


Рисунок 4 - категория создана

Теперь создадим подкатегорию. Чтобы создать подкатегорию к категории “Тест1”, надо нажать на эту категорию в списке, потом ввести название в поле подкатегории и создать.

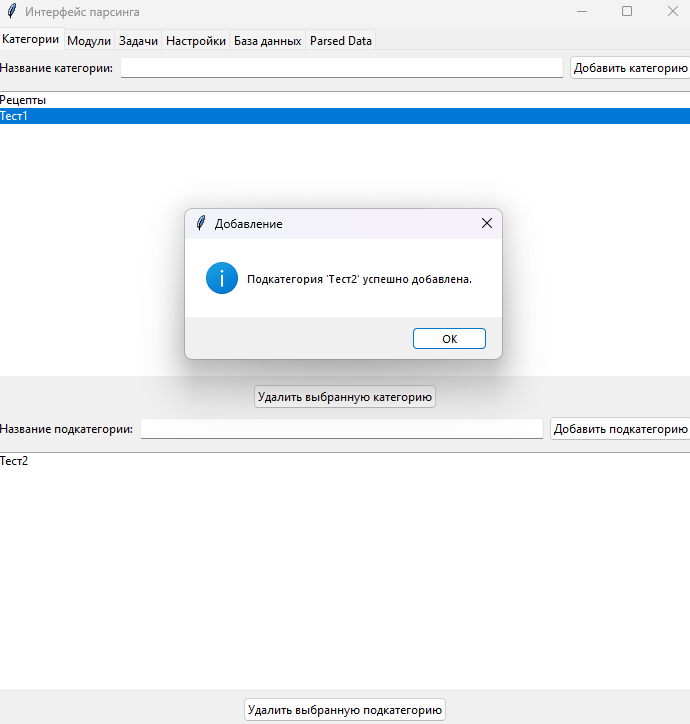


Рисунок 5 - подкатегория создана.

Успешно создана подкатегория “Тест2”. Если удалить категорию “Тест1”, то будут удалены все подкатегории этой категории. Перед тем как перейти к новой вкладки создадим также еще одну подкатегорию - “Тест3”. После успешного создания, перейдем в вкладку “Модуль”. Здесь мы можем подключить модули парсинга, а также дополнительно настроить время запуска. Создадим первый модуль с категории “Тест1” и подкатегории “Тест2”. Также создадим второй модуль с категории “Тест1” и подкатегории “Тест3”.

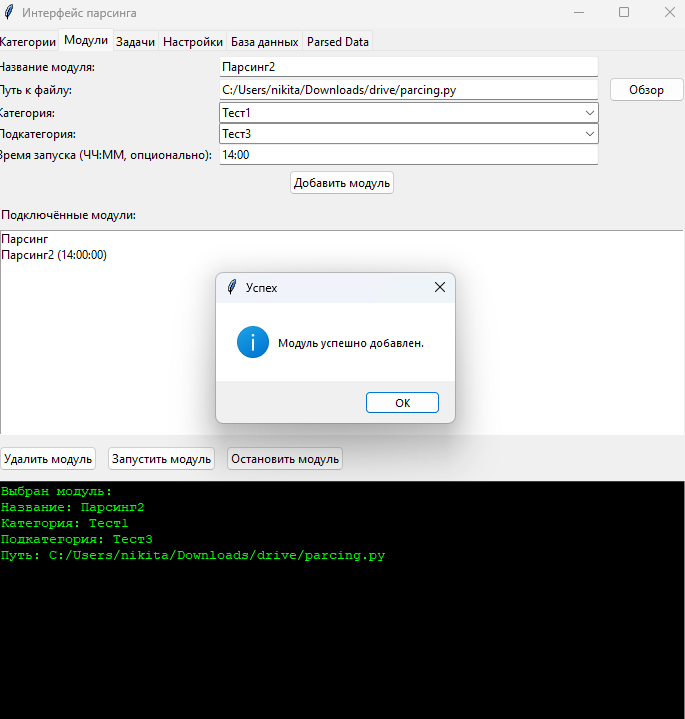


Рисунок 6 - 2 модуля

Успешно создано 2 модуля. “Парсинг2” также имеет время запуска(надо учитывать что модуль запуститься когда ваши часы на компьютере будут 14:00). Можно запустить модуль, остановить или удалить. После успешного парсинга можно увидеть результаты.

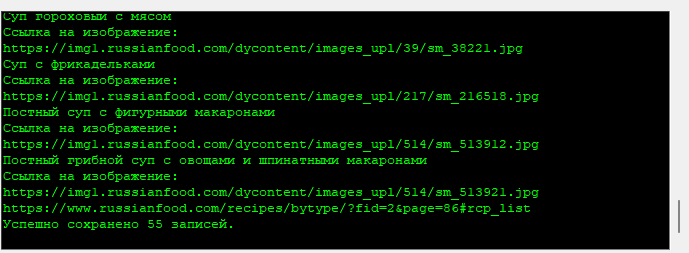


Рисунок 7 - консоль вывода

В консоли вывода хорошо видно все вывода команды print(). Примечание: чтобы ваш код парсинга мог работать с нашим кодом интерфейса у вас не должны быть библиотеки которых нет в списке.

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from webdriver\_manager.chrome import ChromeDriverManager

from bs4 import BeautifulSoup

from datetime import datetime

import pytz

import time

import re

Также ваш код должен сохранять записанные данные в кортеже data, а первые четыре параметра: название, описание, картинка(ссылка) и ссылка на ресурс должны иметь названия data и пронумерованы как на примере(в то же время если вы не хотите парсить или нет возможности иметь один из четырех первых параметров, то это не страшно. Главное чтобы каждый раздел такие как название имели свою нумерацию как data1). Пример:

data.append({

'data1': title, # название

'data2': description, # описание

'data3': image\_url, # картинка

'data4': link, # ссылка

'sub\_description': sub\_description,

'current\_date': current\_date,

'status': 0

})

Также весь ваш код парсинга должен работать в функции def parcing() и возвращать return data. Без этих обязательных настроек ваш код не сможет функционировать с нашим “Модулем”.

Теперь перейдем в вкладку “Задачи”. В нем мы можем создать задачу вывода в телеграмм чат. Для этого надо выбрать нужную вам категорию, подкатегорию(в которой вы запарсили такие то данные), выбрать режим вывода(рандом или по порядку), выбрать эмодзи(из списка, которые заранее созданы или свой создать, изучив идентификатор такого то эмодзи из файла emoji\_list.txt), выбрать время начала и время конца работы вашей задачи(учтите, что надо выбирать время работы с учетом времени на вашем компьютере).

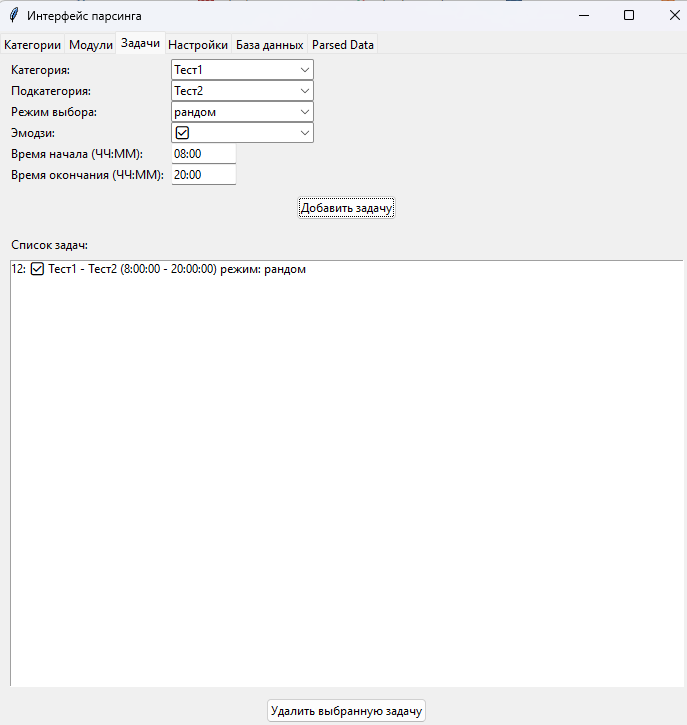


Рисунок 8 - создана задача

Теперь создав задачу перейдем в вкладку “Настройки”. Здесь мы запускаем наш бот вставив его токен в поле. После чего можно нажать на кнопку “Получить список чатов”.

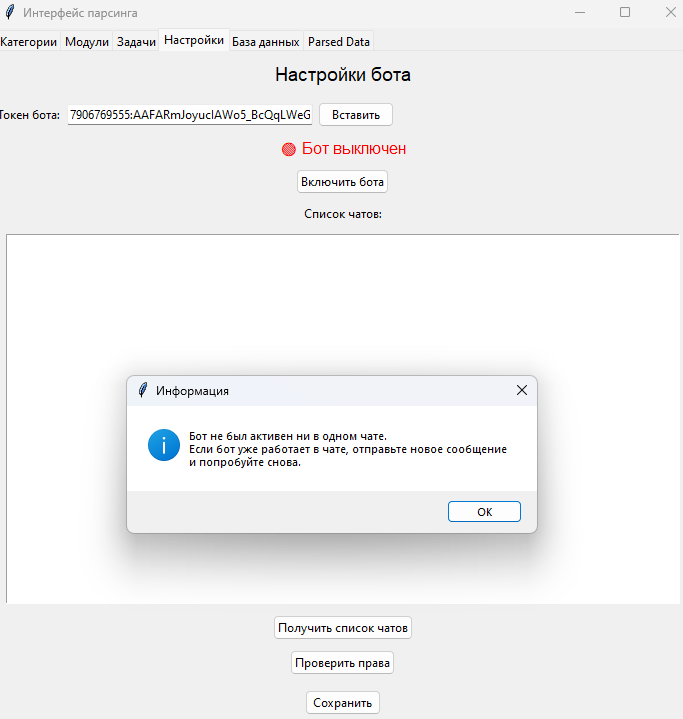


Рисунок 9 - неуспешный поиск чатов

Бывает такое, что чаты не находяться по причине того что вы впервые создали чат или же долго не пользовались системой, а в чате долго никто ничего не писал. Для этого просто отправьте любое сообщение чат и вы получите список чатов.

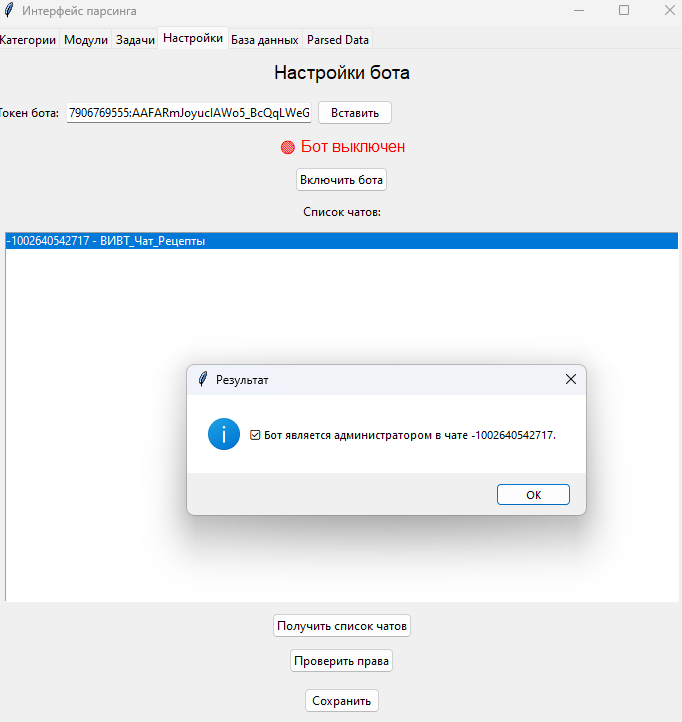


Рисунок 10 - успешный поиск и проверка

После того как нашли чат, вы можете через кнопку “Проверить права” узнать является ваш бот администратором или нет(в зависимости от прав бота, чат может иметь другой id). После чего вы выбираете такой то чат и нажимаете “Сохранить”.

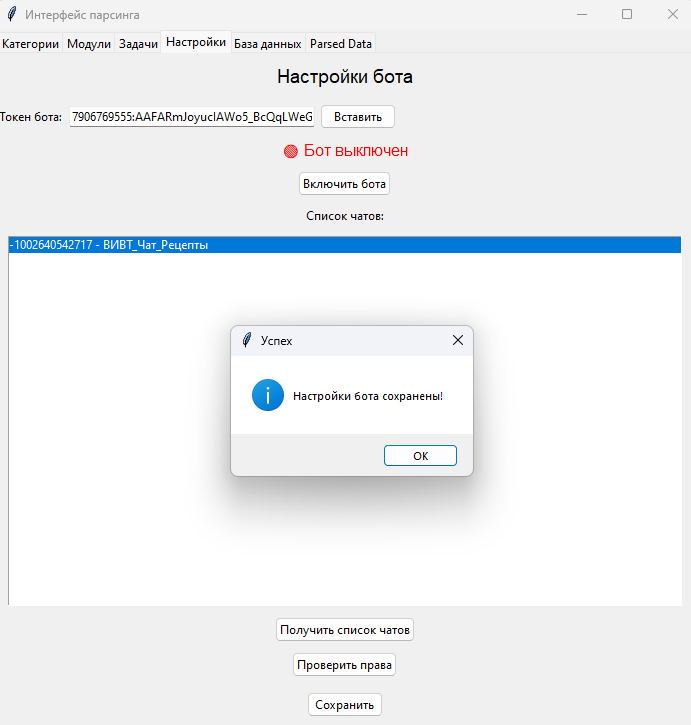


Рисунок 11 - успех выбора чата

Теперь ваши задачи будут выводиться в чат. Выбрав чат включите бота(учтите что первые 2-3 секунды бот будет игнорировать сообщения в чате и ничего не выведет). После запуска и создания задачи в ваш чат после 3 любых сообщений будет выводиться ваши данные согласно настройкам в вашей задаче.

Перейдем в вкладку “База данных”. Здесь вы можете в любой момент поменять базу данных на любую другую(с учетом того, что ваша база MySQL).

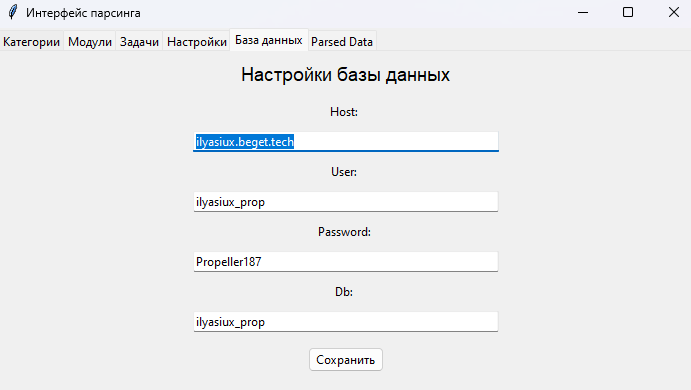


Рисунок 12 - смена базы данных

Наконец последняя вкладка “Parsed Data”. Здесь вы можете сами изменять базу данных, а точнее таблицу в котором сохраняются запарсенные данные по своему усмотрению. Вы можете удалить такую то строчку с данными, удалить все данные или же добавить самим. Вы выбираете категорию и подкатегорию в которую хотите добавить такую то данную, и можете ввести название, описание, ссылка на картинку и ссылка на продукт. Если какие то данные вы не хотите заполнять, то можете пропускать. Даже с одним названием произойдет сохранение.

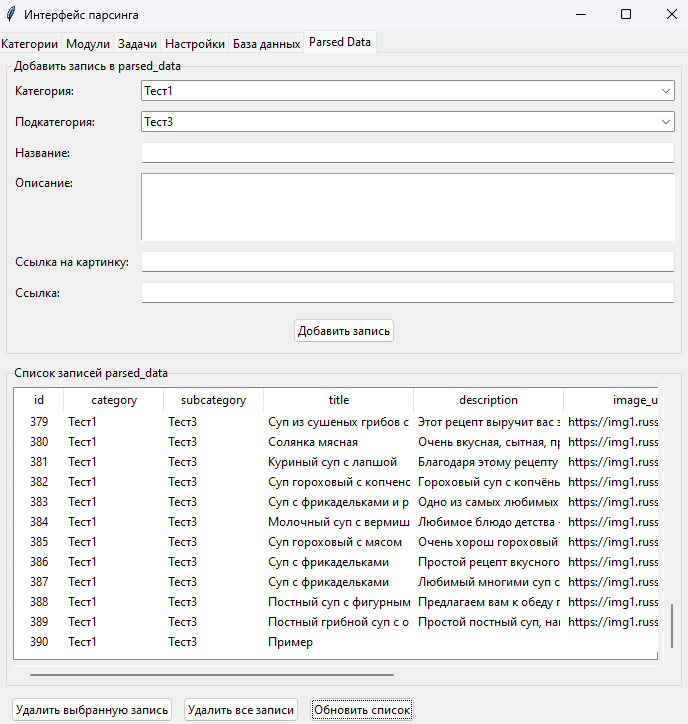


Рисунок 13 - добавлена запись с названием “Пример”

## **Разработка руководства пользователя и администратора**

Для перехода в чат телеграмм можно перейти по qr коду:

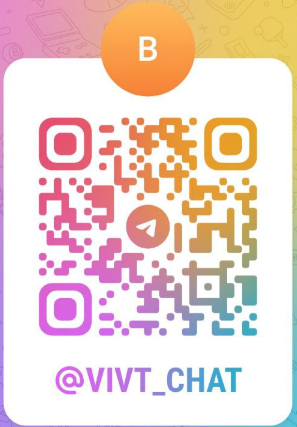


Рисунок 14 - qr код чата

Или же перейти по ссылке: [Вивт-чат](https://t.me/Vivt_chat)

Переходим в чат и начинаем писать три разных сообщения.

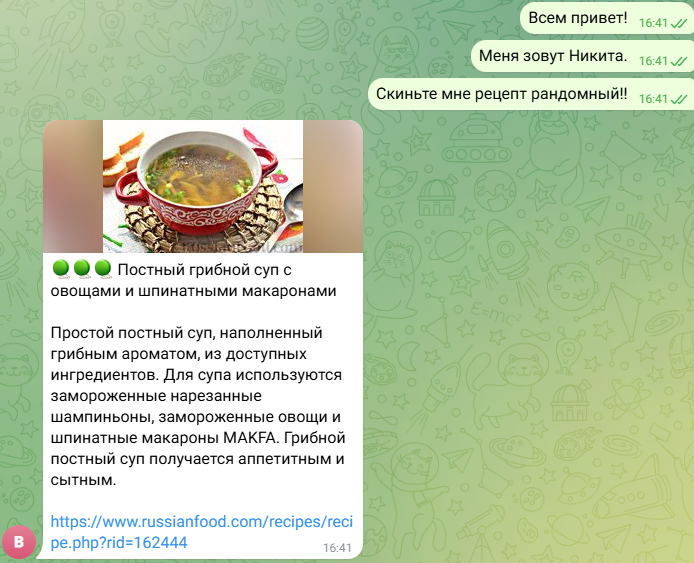


Рисунок 15 - вывод рецепта после 3 сообщений

Как видно, после 3 сообщений в чат выводиться рандомный рецепт из базы данных.