

Хабр



КАК СТАТЬ АВТОРОМ



Хабр торт?



Войти

Актуальные
зарплаты в IT

110k

100k

150k

120k

180k



empenoso

5 июн 2023 в 03:17

Автоответчик на запросы доступа к файлам в гугл почте

👁️ Простой 🕒 3 мин 👁️ 2K

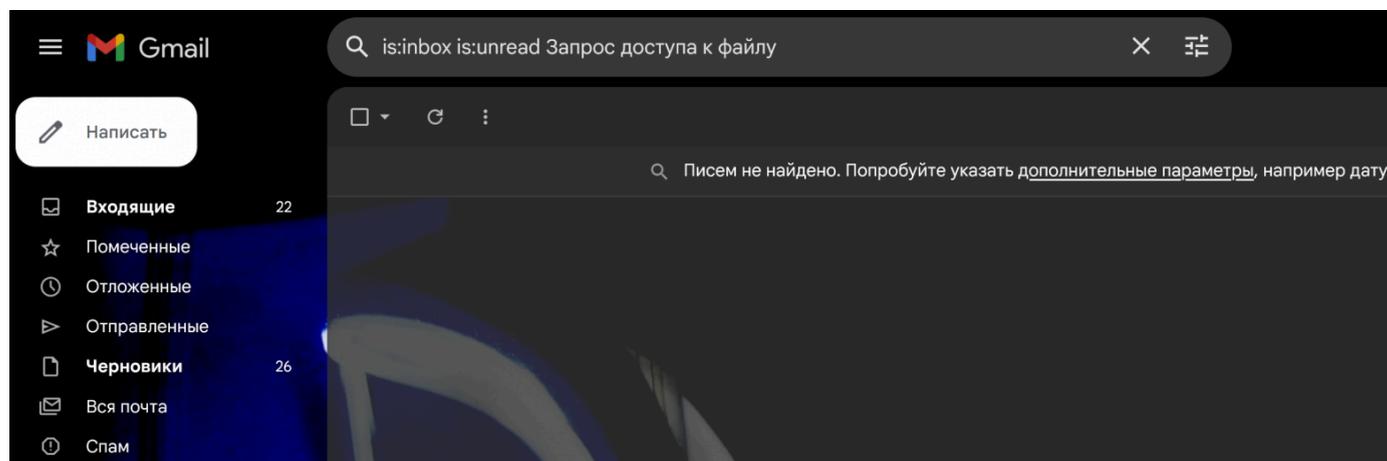
Open source*, JavaScript*, Google API*

Кейс

Недавно мне потребовалось сделать так, чтобы при запросе доступа к файлу на гугл диске через почту отправлялся автоответ. С помощью гугл скриптов я разобрался как это сделать и хотел бы показать это на примере в этой статье.

Как это работает: гугл скрипт ищет непрочитанные электронные письма с запросом на доступ к файлу и отправляет по почте автоответ один раз в сутки, после этого письмо-запрос вместе с автоответом перемещаются в корзину. Если человек напишет вам в эту цепочку, то письмо окажется во входящих.

Для того чтобы понять на какие письма будет приходить автоответ [сделайте запрос в почте](#)
`is:inbox is:unread Запрос доступа к файлу`

`is:inbox is:unread Запрос доступа к файлу`

На моём скриншоте ничего нет, потому что все файлы уже были обработаны скриптом.

Создайте гугл скрипт

Вот шаги для создания скрипта Google Apps, который будет рассылать авто ответы на запросы доступа к файлам:

1. Откройте [Google Диск](#) и выберите «Создать» > «Ещё» > «Google Apps Script». Это откроет редактор скриптов.
2. В редакторе скриптов удалите код по умолчанию и добавьте следующее:

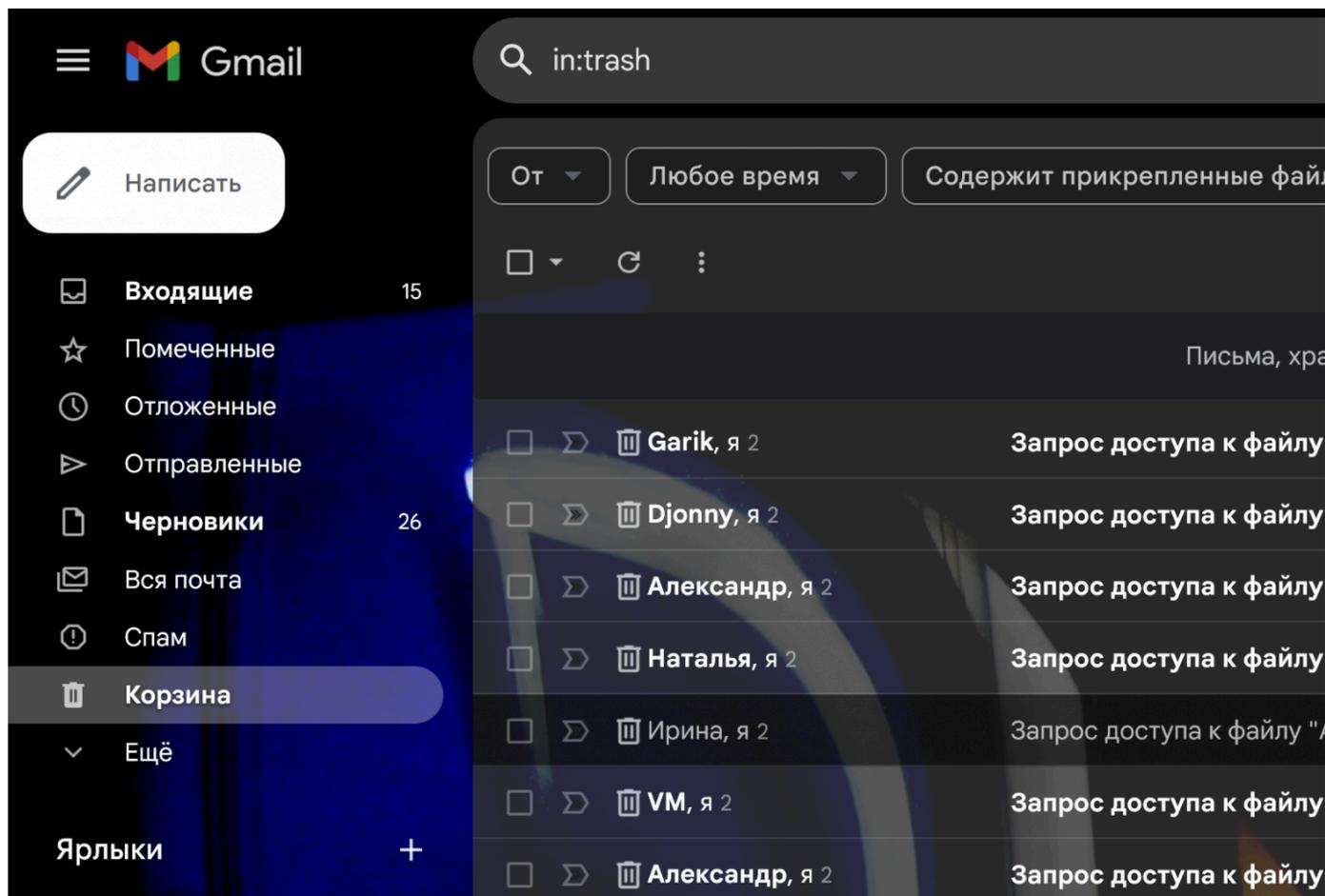
```
// Подробное описание в статье https://habr.com/ru/articles/739168/
// Михаил Шардин, https://shardin.name/

function autoReply() {
  for (const filename of [
    `Имя файла 1`,
    `Имя файла 2`,
    `Имя файла 3`
  ]) {
    var threads = GmailApp.search(`is:inbox is:unread Запрос доступа к файлу "${filename}`);
    for (var i = 0; i < threads.length; i++) {
      var thread = threads[i];
      console.log(`Результат поиска: "${filename}": ${i} из ${threads.length}: \n-
      from = thread
        .getMessages()[thread.getMessageCount() - 1].getFrom()
        .split(' (через)')[0]
        .replace(/\\/g, '')
      thread.reply(`Привет, ${from}!\nБлагодарю Вас за интерес к моему файлу "${filename}`);
      GmailApp.moveThreadsToTrash(threads.slice(i, i + 1));

      console.log(`Был обработан и удалён запрос от ${thread.getMessages()[thread.getMessageCount() - 1].getFrom()}`);
    }
  }
}
```

3. Сохраните сценарий, выбрав «Файл» > «Сохранить». Назовите скрипт, например, «Gmail auto-responder».

4. Запустите сценарий, щелкнув Выполнить, выбрав `autoReply`. Это отправит письма на адреса тех кто запрашивал доступ к файлам, но только если у вас уже есть запросы, то есть `запрос в почте is:inbox is:unread` Запрос доступа к файлу не пустой.



Результат работы скрипта - автоответ отправлен, а затем цепочки с запросами и автоответы перемещены в корзину

3. Добавьте триггер для автоматического запуска скрипта. Перейдите в «Триггеры текущего проекта». Нажмите «Добавить новый триггер» и выберите «Управляемый временем», чтобы запускать скрипт например один раз в день или даже раз в несколько минут.

Изменение триггера для проекта "Gmail auto-respon..."

Выберите функцию	Настройки уведомлений об ошибках
<input type="text" value="autoReply"/>	<input type="text" value="Каждый день"/>
Выполняемое развертывание	
<input type="text" value="Основное развертывание"/>	
Выберите источник мероприятия	
<input type="text" value="Триггер по времени"/>	
Выберите тип триггера	
<input type="text" value="По дням"/>	
Выберите время	
<input type="text" value="С 10:00 до 11:00"/>	
(GMT+05:00)	

Изменение триггера для проекта

Составьте текст письма для автоматического ответа

Это важная часть и я уделил этому достаточно большое место в статье, ведь получатель увидит только её. Можно руководствоваться несколькими правилами:

1. Приветствуйте отправителя по имени, если оно указано.

Вы можете получить имя отправителя из сообщения Gmail, используя: `var`

```
senderName = message.getFrom().split("<")[0].trim();
```

2. Поблагодарите отправителя за запрос на доступ к файлу. Например: Спасибо за ваш запрос на доступ к файлу.
3. Укажите, что это автоматический ответ. Например: Это автоматический ответ системы запросов на доступ к файлам.
4. Укажите предполагаемые сроки ответа на запрос. Например: Мы стараемся отвечать на все запросы на доступ к файлам в течение 2 рабочих дней.
5. Сообщите отправителю, чего ожидать в полном ответе.
6. Еще раз поблагодарите отправителя за терпение и понимание. Например: Еще раз спасибо за ваше терпение и понимание.
7. Заканчивайте вежливо и профессионально. Например: С наилучшими пожеланиями, [Ваше имя]

Заключение

Вот несколько способов использования скрипта Google Apps для автоматического ответа на запросы доступа к файлам по электронной почте:

1. Скрипт может проверять наличие новых непрочитанных писем, содержащих строку `Запрос доступа к файлу` и автоматически отправлять ответ. Это избавляет от необходимости вручную проверять каждый запрос и отвечать на него.
2. Он может заполнять электронное письмо с автоматическим ответом деталями, характерными для каждого запроса, такими как имя отправителя, дата запроса и т. д. Это делает каждый ответ более персонализированным.

Автор: [Михаил Шардин](#),

5 июня 2023 г.

Теги: [автоответчик](#), [автоответы](#), [запрос доступа](#)

Хабы: [Open source](#), [JavaScript](#), [Google API](#)

Редакторский дайджест

Присылаем лучшие статьи раз в месяц



Электропочта



179

30.4

Карма

Рейтинг

Михаил Шардин @empenoso

Разработчик

Подписаться

[Сайт](#) [Сайт](#) [Github](#)

Комментарии 10

Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ

**Erwinmal**

5 часов назад

Кто поджёт Лос-Анджелес? Свежая конспирология о виноватых НЛО, Пи Дидди, урбанистах и корюшке



Простой



14 мин



3.5K

Обзор



+26



4



33

**DimDimDimDimDim**

6 часов назад

Rust 1.84: новый релиз отличного языка программирования. Еще лучше, еще эффективнее, как всегда



6 мин



2.3K



+17



8



2

**JBFW**

14 часов назад

Подключаем длинную линию 1-wire к Ардуино

3 мин

4.2K

+17

32

27

**arturdumchev**

1 час назад

Заговор разработчиков против корпораций

Средний

15 мин

1.2K

Мнение

+11

4

1

**DENEVGSTAR**

5 часов назад

Распознавание образов в мозге с помощью микроплееров

Средний

8 мин

970

Из песочницы

+11

16

5

**chlorine**

7 часов назад

Кэш. Теория кэширования. Устройство и разновидности кэша

Простой

7 мин

1.9K

Из песочницы

+11

64

16

**mikhailmurzak**

21 час назад

Делаем Телеграм-бота в Cursor AI без знания кода

Простой

5 мин

6.5K

Тutorial

+11

75

15



burenkov

3 часа назад

Стереокамера машинного зрения с поддержкой ИИ на базе FPGA и Arduino Portenta H7

10 мин 712

Из песочницы

+10

11

0



andreybold

7 часов назад

Как я развлекался с восходом солнца

Простой 6 мин 1.4K

+7

5

30



AlexandrovAndrey

21 час назад

Go: тонкости профилирования CPU

Простой 16 мин 1.3K

Tutorial

+7

20

2

Роботяги: что будет делать ИИ в промышленности к 2035 году

Турбо

Показать еще

ИСТОРИИ



Годнота из блогов компаний



Выравнения планет



Нейрозима 2025



Статьи с новогодним вайбом



Кто выступит на конференции мечты

ВОПРОСЫ И ОТВЕТЫ

CME IVR TCL скрипт, как изменить поведение?

Cisco · Сложный · 1 ответ

Телефонный автоответчик через api?

API · Средний · 0 ответов

Сотовый автоответчик, мобильный телефон с функцией автоответчика - существует ли?

Мобильные устройства · Средний · 1 ответ

Как сделать автоответчик по расписанию в Viber?

Автоматизация · Сложный · 0 ответов

Кто объяснит аргументированно про невозможность голосового автоответчика в Android?

Android · Средний · 2 ответа

[Больше вопросов на Хабр Q&A](#)

МИНУТОЧКУ ВНИМАНИЯ



Разгребаем задачи после праздников с task-трекером



Иди со мной, если хочешь на перекур: будущее ИИ на заводах



Чтобы победить Переработу, нужно всего лишь...

РАБОТА

[JavaScript разработчик](#)

98 вакансий

[React разработчик](#)

30 вакансий

[Все вакансии](#)

БЛИЖАЙШИЕ СОБЫТИЯ



30 января

Зимний тест-драйв Хабра для компаний

Москва

Маркетинг

Другое

[Больше событий в календаре](#)

Хабр



 [Настройка языка](#)

[Техническая поддержка](#)

© 2006–2025, Habr