



Все важнейшие
IT-события здесь

5

6

7

8

9

10

11

12

13

14

15

16

17

18



empenoso

14 апр 2023 в 03:17

Как создать множество документов на основе единого шаблона при помощи скрипта внутри гугл таблицы

Средний

6 мин

4.4K

Open source*, JavaScript*, API*, Google API*

Кейс

Иногда бывает необходимо создать множество повторяющихся документов, которые отличаются лишь номером, датой и ещё парой текстовых строк. Очень грустно тратить на их создание своё время - ведь требуется совершить множество одинаково повторяющихся действий. Ещё можно понять затраты времени на создание 5 документов, но если их надо создать, например 500 штук?

А ещё эта задача может усложниться за счёт необходимости вставки каких либо данных, которые надо сначала найти в интернете. Например, вставлять данные о погоде на дату создания документа.

К счастью, гугл таблицы могут упростить процесс создания однотипных копий с помощью шаблона. В этой статье покажу как можно создать неограниченное число копий на основе одного шаблона с помощью гугл скрипта внутри гугл таблицы.

Создайте шаблон для вашего документа

Первым шагом является создание шаблона. Пусть образцом для этой статьи станет протокол сварки. Его можно создать заново или перенести в гугл таблицы. [Вот ссылка на этот пример.](#)

	А	В	С
1	Погодные условия	Ниже	
2	Окружающая температура, °С	ТЕМПЕРАТУРА	
3	/=====/		
4	Оператор	СВАРЩИК	
5	Номер шва	НОМЕР	
6	Регистратор	Nowatech	
7	Серийный номер регистратора	15795	
8	Сварочный аппарат	Nowatech	
9	Серийный номер	15795	
10	Версия	2	
11	Порог давления Т1 Т2	2.0 бар	
12	Порог давления Т2 Т3	0.5 бар	
13	/=====/		
14	Диаметр трубы, мм	110	
15	SDR	11	
16	Толщина стенки, мм	10	
17	Материал	100	
18	/=====/		
19	Параметр	Заданный	Фактический
20	/=====/		
21	Давление движения, бар	--	
22	Высота грата, мм	3.0	
23	Температура, гр	220.0	
24	Давление сварки Р1(Р1), бар + ДД	38.6 (19.3)	
25	Время нагрева Т1, сек	--	
26	Время нагрева Т2, сек	286	
27	Время паузы Т3, сек	13	
28	Время увеличения давления Т4, сек	16	
29	Давление сварки Р5(Р5), бар + ДД	38.6 (19.3)	
30	Время охлаждения Т5, мин(сек)	34(2040)	
31	Общее время, мин (сек)	40 (2400)	
32	/=====/		
33	Начало записи	6 октября 2022 г.	08:42
34	Конец записи	6 октября 2022 г.	09:30
35			
36			

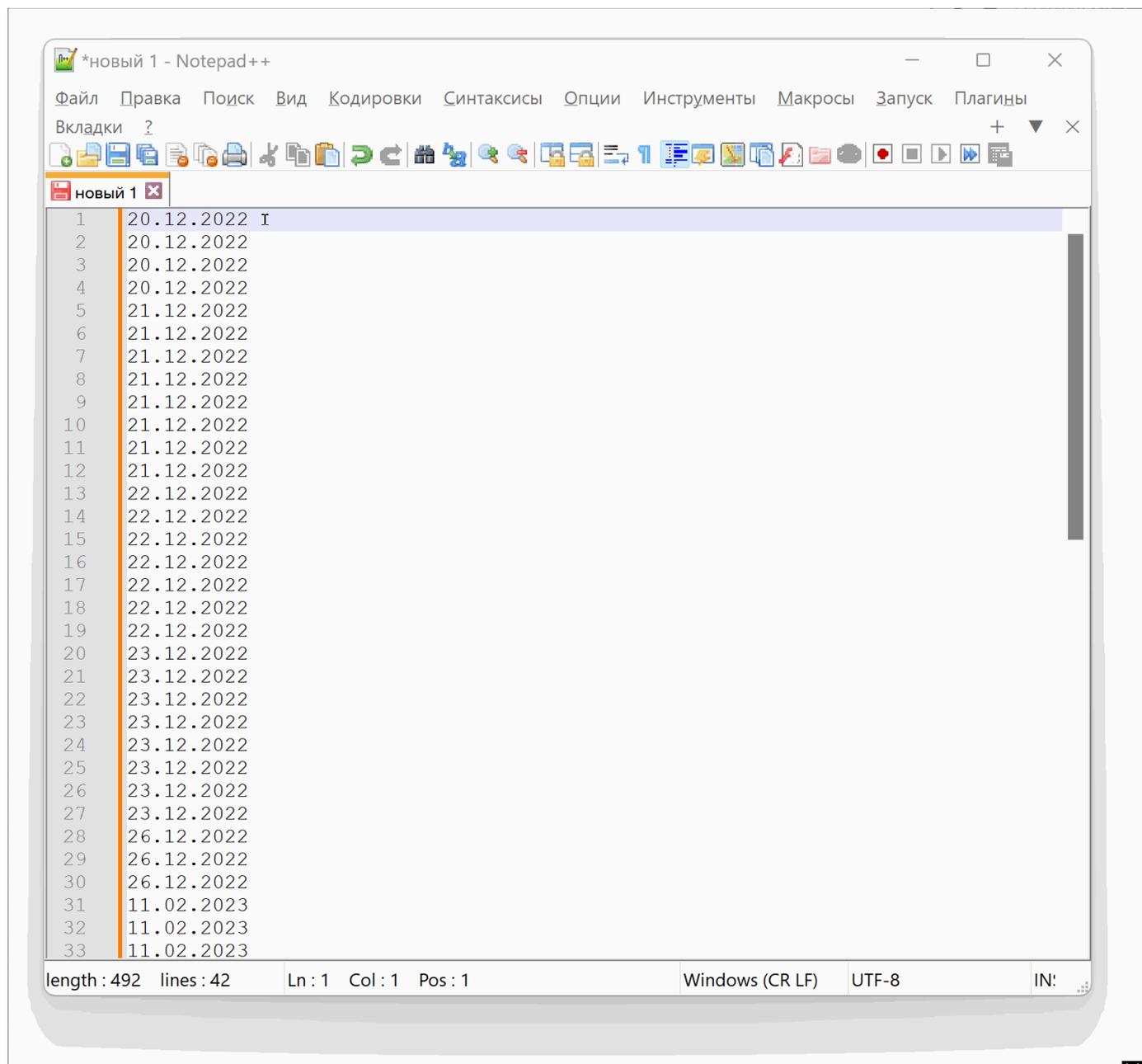
Шаблон протокола на одну страницу А4. Красным выделены поля, которые надо изменять

Красным выделены поля, которые надо изменять в процессе создания.

Даты были заданы заранее, как и их порядковые номера. Даты для работы скрипта надо записать в виде массива:

```
const values = ["2022-12-21", "2022-12-21", "2022-12-21", "2022-12-21", "2022-12-21", "2022-12-21", "2022-12-21", "2022-12-22", "2022-12-22"];
```

Для столбца из дат можно сделать это, например, через макросы в Notepad++:
(знаете способ лучше - напишите в комментарии к статье)



Запись макроса и его дальнейший запуск

На гифке показано как сделать запись макроса в Notepad++ и его дальнейший запуск до конца файла, для того чтобы сделать из обычного столбца дат массив в гугл скриптах.

Гугл скрипты в помощь

Google Apps Script — это простой язык сценариев, который использует синтаксис JavaScript.

Для получения погоды (а это одно из полей, которые требует протокол) можно воспользоваться бесплатным [Free Weather API](#), которое даже не требует получения ключа для работы с ним. Координаты местности задаются через параметры

```
latitude=58.08&longitude=55.76 :
```

```
// https://habr.com/ru/articles/728840/  
// Михаил Шардин https://shardin.name/  
  
function temperature(date) { //получаем погоду  
  // date = "2022-10-04"  
  // console.log(`temperature. date = ${date}`)  
  const url = `https://archive-api.open-meteo.com/v1/archive?latitude=58.08&longitude=55.76&date=${date}&timezone=Europe/Moscow`  
  CacheServis(url);  
  try {  
    const response = UrlFetchApp.fetch(url)  
    const json = JSON.parse(response.getContentText());  
    const temperature_2m_mean = json.daily.temperature_2m_mean[0]  
    // console.log(`Погода по координатам на ${date}: ${temperature_2m_mean}°C.`)  
    const precipitation_sum = json.daily.precipitation_sum[0]  
    // console.log(`Погода по координатам на ${date}: ${precipitation_sum} мм.`)  
    const windspeed_10m_max = json.daily.windspeed_10m_max[0]  
    // console.log(`Погода по координатам на ${date}: ${windspeed_10m_max} км/ч.`)  
    // console.log(`Погода по координатам на ${date}: ${temperature_2m_mean}°C, ${precipitation_sum} мм осадков, ${windspeed_10m_max} км/ч.`)  
    return `${temperature_2m_mean}°C\n${precipitation_sum} мм осадков\nветер ${windspeed_10m_max} км/ч`  
  } catch (error) {  
    console.log(`temperature. Ошибка на ${date}: ${error}.`)  
    return ""  
  }  
}
```

Поскольку даты повторяются пришлось воспользоваться сервисом кеширования, для того чтобы сократить время работы этой погодной функции:

```
// https://habr.com/ru/articles/728840/  
// Михаил Шардин https://shardin.name/
```

```
function CacheServis(url) { //кэширование результата запроса по url
  var cache = CacheService.getScriptCache();
  var cached = cache.get(url);
  if (cached != null) {
    // Logger.log("CacheServis.\nУже было записано для " + url);
    return cached;
  }
  try {
    var result = UrlFetchApp.fetch(url);
  } catch (error) {
    Logger.log("CacheServis.\nОшибка обращения при кэшировании по url " + url + "\n");
    return ""
  }
  var contents = result.getContentText();
  cache.put(url, contents, 21600); // cache for 6 hours
  Logger.log("CacheServis.\nЗаписали по новой для " + url);
  return contents;
}
```

Далее самая важная часть гугл скрипта - создание множества листов на основе одного шаблона:

```
// https://habr.com/ru/articles/728840/
// Михаил Шардин https://shardin.name/

function index() {
  var startTime = new Date();

  const values = ["2022-12-21", "2022-12-21", "2022-12-21", "2022-12-21", "2022-12-21"];

  let n = 0;
  let time = 7;

  values.forEach((date, index) => {
    if (index === 0 || date !== values[index - 1]) {
      time = 7;
    } else {
      time += 1;
    }
    n += 1;
  });
}
```

```
    console.log(`Выполняю копирование №${n} для ${date}.`)  
    duplicate(date, n, time);  
  
    var endTime = new Date();  
    var duration = (endTime - startTime) / 60000;  
    Logger.log("Время выполнения скрипта: " + duration.toFixed(2) + " минут");  
  });  
}
```

И сам код, который создаёт дубликаты вкладок и меняет их содержимое:

```
// https://habr.com/ru/articles/728840/  
// Михаил Шардин https://shardin.name/  
  
function duplicate(date, n, time) {  
  
    var ss = SpreadsheetApp.getActiveSpreadsheet();  
    var lastTabName = ss.getSheets().pop().getSheetName();  
    var sheet = ss.getSheetByName('Шаблон').copyTo(ss);  
  
    sheet.setName(n);  
    sheet.getRange("B2").setValue(temperature(date));  
    sheet.getRange("B5").setValue(n);  
    sheet.getRange("B33").setValue(date);  
    sheet.getRange("B34").setValue(date);  
    sheet.getRange("C33").setValue(`${time}:${Math.floor(Math.random() * 3) + 1}:00`);  
    sheet.getRange("C34").setValue(`${time+1}:${Math.floor(Math.random() * 6) + 4}:00`)  
  
    // ss.setActiveSheet(sheet);  
}
```

Я знаю что этот код написан не самым оптимальным образом и вряд ли стоит рассчитывать на скорость работы, но этот код может за 6 бесплатных минут работы гугл скрипта (а именно такое ограничение любого гугл аккаунта), создать около 300 дубликатов вкладки. А ещё обращение к внешнему сервису за погодой сильно замедляет его работу.

Как пользоваться?

Для того чтобы запустить этот скрипт из гугл таблицы и предоставить все необходимые разрешения для его работы надо выполнить несколько шагов:

Шаг 1. Откройте таблицу Google `Протоколы_шаблон` и скрипт :

- Войдите в свою учетную запись Google и откройте электронную таблицу, содержащую этот скрипт.

Шаг 2. Создайте собственную копию этой гугл таблицы примера:

- Нажмите на меню «Файл», затем выберите «Создать копию...».
- В окне «Создать копию» введите имя для своей копии электронной таблицы и выберите место для нее.
- Нажмите «ОК», чтобы создать копию.
- Ваша новая копия электронной таблицы откроется в новой вкладке, теперь вы можете начать работать с ней.
- Все данные и форматирование исходной электронной таблицы будут перенесены в новую копию.
- Если вы сделали копию общей электронной таблицы, у вас будут свои собственные отдельные данные, и вы не испортите данные исходной электронной таблицы, которой с вами поделились.

Шаг 3. Откройте редактор в вашей таблице скриптов:

- Щелкните меню «Расширения», затем выберите «Apps Script».
- Это откроет редактор скриптов в новом окне.

Шаг 4. Запустите скрипт:

- В редакторе сценариев выберите вкладку `replication`, а в ней функцию `index` и щелкните кнопку «Выполнить».
- Вам потребуется авторизовать сценарий, нажав «Просмотреть разрешения», а затем нажав «Разрешить», чтобы предоставить сценарию разрешение на доступ к информации вашей учетной записи для того, чтобы начать использовать этот скрипт. Вы в любое время можете посмотреть список выданных вами [разрешений на специальной странице](#) и в один клик их отозвать.

Шаг 5. Проверьте результат:

- Если сценарий выполняется успешно, выходные данные функции будут отображаться на вкладке «Журнал выполнения» в нижней части окна редактора сценариев.

Журнал выполнения		✕
15:03:55	Примечание	Выполнение начато
15:03:56	Информация	Выполняю копирование №1 для 2022-12-21.
15:03:57	Информация	Время выполнения скрипта: 0.02 минут
15:03:57	Информация	Выполняю копирование №2 для 2022-12-21.

Выполнение работы скрипта

- Также во время работы скрипта вы можете открыть таблицу и наблюдать как появляются новые вкладки.

Вот и всё! Следуя этим простым шагам, вы можете запустить этот гугл скрипт из своей электронной таблицы и убедиться, что всё работает. После этого можно скачать готовую книгу как файл Экселя или pdf и передать её в дальнейшую работу.

25	Время нагрева T1,сек	--	
26	Время нагрева T2,сек	286	
27	Время паузы T3,сек	13	
28	Время увеличения давления T4,сек	16	
29	Давление срабки P5(P5) бар ± ПП	38.6 (10.3)	

Вновь созданные вкладки

Дополнительный шаг. При помощи функции `deleteRedundantSheets` можно удалить все автоматически созданные листы, для того чтобы подготовить этот шаблон уже с другими параметрами.

Что в итоге

Использование связки гугл таблица + гугл скрипт для создания однотипных документов на основе единого шаблона является действенным способом автоматизации создания документов.

Этот процесс поможет сэкономить время и повысить производительность. С помощью пошагового руководства, представленного в этой статье, вы сможете легко настроить этот процесс самостоятельно, используя связку гугл таблица + гугл скрипт.

Настраивая свои шаблоны и сценарии в соответствии со своими потребностями, вы сможете создавать свои собственные профессионально выглядящие документы.

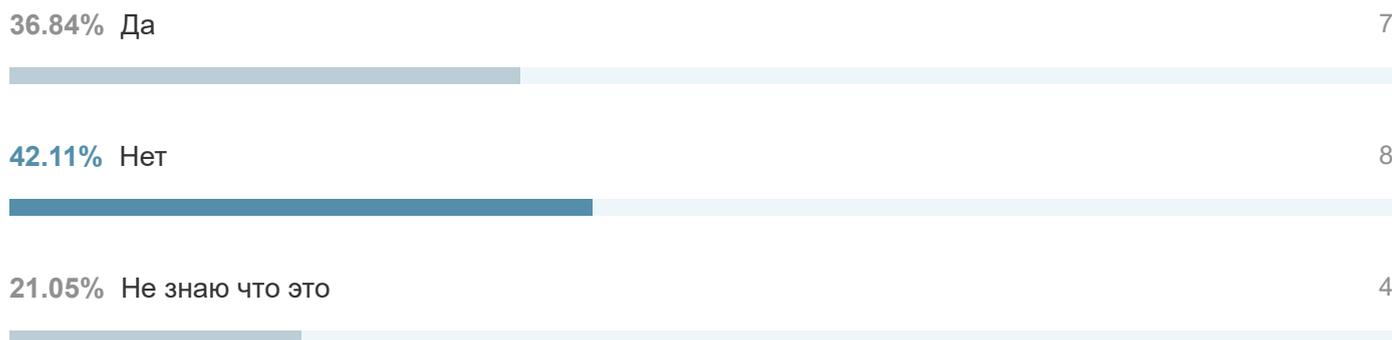
После того, как все будет работать хорошо, вы сможете воспользоваться преимуществами оптимизированного процесса создания документов и сосредоточиться на самом важном — легком и эффективном достижении ваших целей.

Автор: [Михаил Шардин](#),

14 апреля 2023 г.

Только зарегистрированные пользователи могут участвовать в опросе. [Войдите](#), пожалуйста.

Используете гугл скрипты в работе?



Проголосовали 19 пользователей. Воздержавшихся нет.

Теги: [шаблон](#), [документ](#), [гугл скрипт](#), [таблицы](#)

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

Редакторский дайджест



Присылаем лучшие статьи раз в месяц



179

30.4

Карма

Рейтинг

Михаил Шардин @empenoso

Разработчик

Подписаться



[Сайт](#) [Сайт](#) [Github](#)

Комментарии 6

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  1.3K +11 4 1**DENEVGSTAR**

5 часов назад

Распознавание образов в мозге с помощью микроплееров

 Средний  8 мин  970 +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



burenikov

3 часа назад

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

 10 мин  714

Из песочницы

 +10

 12

 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

Тutorial

◆ +7

📄 20

💬 2

Рабочая схема: делимся рейтингом работодателей в IT за 2024 год

Турбо

Показать еще

ИСТОРИИ



Годнота из блогов компаний



Выравнения планет



Нейрозима 2025



Статьи с новогодним вайбом



Кто выступит на конференции мечты

СРЕДНЯЯ ЗАРПЛАТА В IT

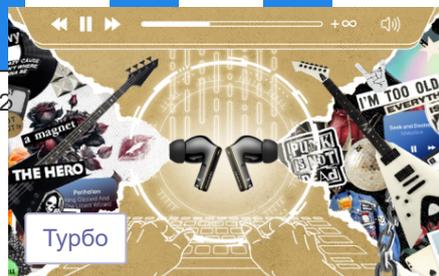
199 942 ₺/мес.

— средняя зарплата во всех IT-специализациях по данным из 8 534 анкет, за 1-ое пол. 2025 года. Проверьте «в рынке» ли ваша зарплата или нет!

МИНУТО ВНИМАНИЯ



Гиперконвергентная среда:
OpenStack или VMware?



Так ли хороши звук и шумодав:
обзор наушников FreeBuds Pro 4



Разгребаем задачи после
праздников с таск-трекером

РАБОТА

React разработчик

30 вакансий

JavaScript разработчик

98 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