



# Free XLS FileService: SQL-доступ к Excel-файлам через Web API

---

**Free XLS FileService** — это бесплатный кроссплатформенный веб-сервис с открытым исходным кодом, который позволяет обращаться к данным из Excel-файлов как к SQL-таблицам через Web API. Сервис идеально подходит для некоммерческого использования, автоматизации, макросов и интеграции Excel-данных в другие системы.

Он поддерживает:

- управление файлами (список, проверка существования, метаданные);
- выполнение SQL-запросов к данным Excel;
- быструю и удобную интеграцию без необходимости запуска Excel.

Free XLS FileService упрощает работу с табличными данными — особенно в сценариях, где требуется простой программный доступ к Excel-файлам через HTTP.



# Что такое Free XLS FileService?

## **Ваши проблемы:**

- Неудобный доступ к данным из Excel-файлов на уровне кода.
- Excel плохо интегрируется с внешними системами, особенно при автоматизации и использовании макросов.
- Ручной перенос данных — риск ошибок, потери времени и ресурсов.

## **Наше решение: Free XLS FileService**

- Бесплатный, кроссплатформенный веб-сервис с открытым исходным кодом.
- Обрабатывает Excel-файлы как SQL-таблицы — данные доступны через стандартные GET-запросы.
- Легко интегрируется в макросы Р7-Офис, скрипты и сторонние приложения.
- Устанавливается за минуту, не требует других платных решений.
- Идеально для разработчиков, интеграторов и всех, кто автоматизирует работу с Excel-данными.



# Установка и запуск сервиса. Вариант 1 – ручной запуск (для разработчиков)

---

Выполните следующие шаги ниже, чтобы запустить Free XLS FileService, \*, \*\*

1. `git clone https://github.com/r7-consult/r7_fileservice.git`
2. `cd r7_fileservice`
3. `pip install -r requirements.txt`
4. `python -m xldb-proxy-file`

\*Требуется установленный Python и Git.

\*\* По умолчанию сервис запускается на `http://127.0.0.1:58081`



## Установка и запуск сервиса. Вариант 2 – запуск из готовой сборки

---

Выполните следующие шаги ниже, чтобы запустить Free XLS FileService, \*

1. Скачайте исполняемый файл для вашей системы:

↓ Ubuntu (`r7\_fileservice-ubuntu.bin`)

↓ RedOS (`r7\_fileservice-redos.bin`)

↓ AstraOS (`r7\_fileservice-astra.bin`)

2. Выполните: `./r7\_fileservice-[os]` (запустите нужный билд)

\* По умолчанию сервис запускается на <http://127.0.0.1:58081>



# Возможности и примеры работы (API Endpoints)

**GET /file/list** - Получение списка файлов в директории

<http://localhost:58081/file/list?path=C:\>

Ответ:

Успех: {"files": ["file1.xlsx", "file2.xlsx", ...]} - список файлов

Ошибка: HTTP 500 с описанием ошибки (например, директория не существует)

**GET /file/available** - Проверка существования файла

[http://localhost:58081/file/available?path=C:\&file\\_name=example.xlsx](http://localhost:58081/file/available?path=C:\&file_name=example.xlsx)

Ответ:

Файл существует: {"status": "OK"}

Файл не существует: {"status": "Error"}

Ошибка: HTTP 500 с описанием ошибки

**GET /file/meta** - Получение метаданных файла

[http://localhost:58081/file/meta?path=C:\&file\\_name=example.xlsx](http://localhost:58081/file/meta?path=C:\&file_name=example.xlsx)

Ответ:

Успех: JSON с метаданными файла:

```
{ "file_name": "example.xlsx", "file_path": "C:\\example.xlsx", "size_bytes": 1234,  
  "created_at": "2025-05-20T12:00:00", "modified_at": "2025-05-20T12:00:00",  
  "accessed_at": "2025-05-20T12:00:00", "mode": "0o100666", "sheets": ["Sheet1",  
  "Sheet2"], "copyright": "Free for non commercial use https://r7-consult.ru/" }
```

Ошибка: HTTP 500 с описанием ошибки



# Возможности и примеры работы (API Endpoints)

**GET /file/register** - Проверка возможности чтения Excel-файла и получение информации о его структуре  
[http://localhost:58081/file/register?path=C:\&file\\_name=example.xlsx&sheet\\_name=Sheet1&usecols=A,B,C&skiprows=0&nrows=1000](http://localhost:58081/file/register?path=C:\&file_name=example.xlsx&sheet_name=Sheet1&usecols=A,B,C&skiprows=0&nrows=1000)

## Параметры:

sheet\_name: имя листа (опционально)  
usecols: список колонок (опционально)  
skiprows: пропустить строк сверху (по умолчанию 0)  
nrows: максимальное число строк (по умолчанию 10000000)

## Ответ:

Успех: {"columns": ["col1", "col2", ...], "rows": 1000} - список колонок и количество строк в файле  
Ошибка: HTTP 500 с описанием ошибки



# Возможности и примеры работы (API Endpoints)

**GET /file/register** - Проверка возможности чтения Excel-файла и получение информации о его структуре

[http://localhost:58081/file/register?path=C:\&file\\_name=example.xlsx&sheet\\_name=Sheet1&usecols=A,B,C&skiprows=0&nrows=1000](http://localhost:58081/file/register?path=C:\&file_name=example.xlsx&sheet_name=Sheet1&usecols=A,B,C&skiprows=0&nrows=1000)

## Параметры:

sheet\_name: имя листа (опционально)

usecols: список колонок (опционально)

skiprows: пропустить строк сверху (по умолчанию 0)

nrows: максимальное число строк (по умолчанию 10000000)

## Ответ:

Успех: {"columns": ["col1", "col2", ...], "rows": 1000} - список колонок и количество строк в файле

Ошибка: HTTP 500 с описанием ошибки

## Примеры запросов

select \* from df -- все данные

select col1, col2 from df -- только нужные колонки

select \* from df where col1 > 100 -- с условием

select col1, count(\*) from df group by col1 -- с группировкой





# Интеграция с Р7-Офис: Макрос для отображения данных

Интеграция Free XLS FileService с Р7-Офис позволяет напрямую выполнять SQL-запросы к данным Excel и вставлять результаты на лист. Макрос отправляет GET-запрос, получает JSON-ответ, преобразует его в таблицу и автоматически обновляет отображение.

## — Отправка GET запроса и получение JSON

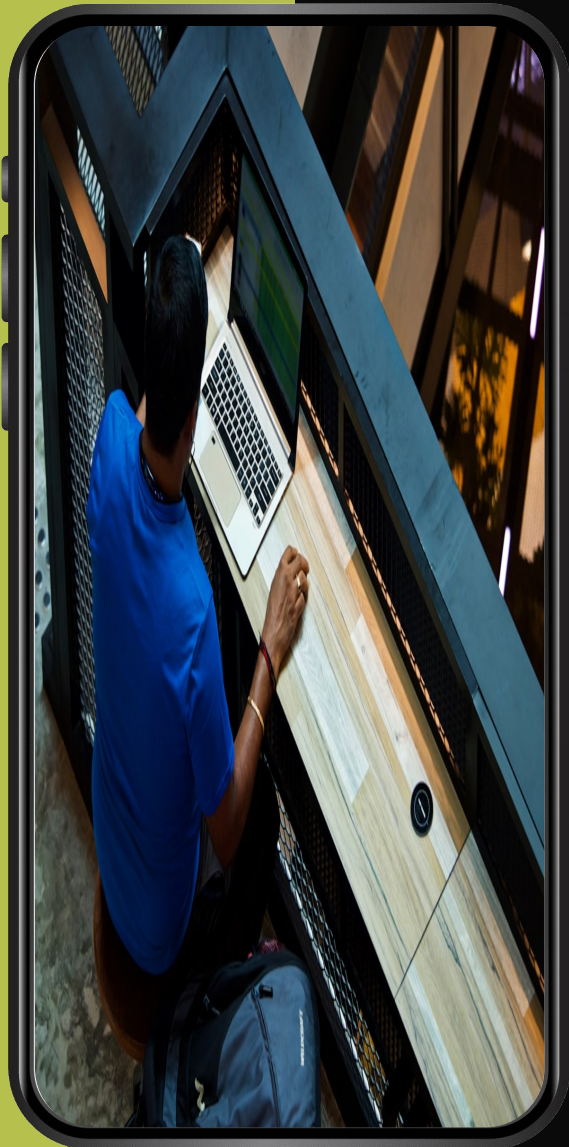
```
var url =  
"http://localhost:58081/file/sql?query=select%20*%20from%20df&path=C:\\tt&file_name=address.xlsx&sheet_name=Sheet1"; //Считываем файл  
fetch(url)  
  .then(response => response.json()) //Парсим ответ как JSON  
  .then(data => {...})
```

## — Формируем данные для вставки, выделяем диапазон

```
// Объединяем заголовки и строки данных в один массив  
var fullArray = [headers, ...sData];  
// Получаем диапазон для всей таблицы  
var dataRange = oWorksheet.GetRange(  
  oWorksheet.GetRangeByNumber(0, 0),  
  {...})
```

## — Вставляем данные

```
dataRange.SetValue(fullArray);  
{...}
```





# Структура проекта (для разработчиков)

Заглянем под капот: Структура проекта представлена деревом каталогов. Основные модули находятся в `src/`, а исполняемые файлы - в `build/`.

```
r7_fileservice/  
├── src/  
│   ├── logger_conf.py  
│   ├── main.py  
│   ├── r7_fileservice.py  
│   ├── reporting.py  
│   ├── svc_xls_files.py  
│   └── types_meta.py  
├── build/  
│   ├── r7_fileservice-windows.exe  
│   ├── r7_fileservice-ubuntu.bin  
│   ├── r7_fileservice-redos.bin  
│   └── r7_fileservice-astra.bin  
├── requirements.txt  
└── README.md
```





## Итоги и Преимущества

**Free XLS FileService** — это ваш легкий путь к SQL-доступу к Excel.  
Работайте с Excel-файлами как с полноценными SQL-таблицами — через простой Web API.

- **Интеграция за минуты:** Идеально подходит для макросов Р7-Офис и любых приложений, которые умеют делать GET-запросы
- **Минимум усилий, максимум совместимости:** Кроссплатформенный и не требует установки Excel — всё работает на стороне сервера
- **Открыто и бесплатно:** Исходный код доступен на GitHub. Бесплатно для некоммерческого использования.
- **Установка в один шаг:** Готовая сборка или запуск через Python — выбирайте то, что удобно вам

Попробуйте прямо сейчас: [github.com/r7-consult/r7\\_fileservice](https://github.com/r7-consult/r7_fileservice)

Сайт: [r7-consult.ru](http://r7-consult.ru)

## Вопросы? / Контакты

Сайт: [r7-consult.ru](http://r7-consult.ru)

Email: [er@exceldb.pro](mailto:er@exceldb.pro)

Телефон: +7 915 258-0371

Telegram: [https://t.me/r7\\_js](https://t.me/r7_js)

GitHub: [https://github.com/r7-consult/r7\\_fileservice](https://github.com/r7-consult/r7_fileservice)

