

MarkItDown: любой файл в Markdown для LLM за одну команду

Скормить модели сырой PDF, Word или Excel напрямую — потерять структуру документа и сжечь лишние токены на форматирование, которое ей не нужно. MarkItDown от Microsoft превращает практически любой файл в чистый Markdown — формат, на котором языковые модели обучены и который они «понимают» нативно.

Важно: MarkItDown делает I/O с правами того процесса, в котором запущен, — как обычный `open()` или `requests.get()`. На недоверенном вводе не вызывай `permissive-метод convert()` напрямую — используй более узкий `convert_local()` или `convert_stream()` и санитизируй пути и URI заранее.

1. Зачем конвертировать в Markdown для LLM

- Markdown почти не отличается от простого текста, но сохраняет важную структуру документа — заголовки, списки, таблицы, ссылки.
- Модели вроде GPT-4o обучены на огромных объёмах текста в Markdown и нативно «говорят» на нём, часто вставляя разметку в ответы без явной просьбы.
- Побочный эффект такой обученности — соглашения Markdown ещё и token-эффективны по сравнению с исходной разметкой файла.

2. Что конвертирует и в цифрах

- **Форматы:** PDF, PowerPoint, Word, Excel, изображения (EXIF-метаданные + OCR), аудио (метаданные + распознавание речи), HTML, CSV/JSON/XML, ZIP-архивы (разбирает содержимое), ссылки на YouTube, EPub и другие.
- 162 тысячи звёзд на GitHub, лицензия MIT, проект развивает команда AutoGen внутри Microsoft.
- Расширяется сторонними плагинами: команда `markitdown --list-plugins` показывает установленные, `markitdown --use-plugins path-to-file.pdf` включает их для конкретного запуска.

3. Установка и запуск из терминала

- **Требование:** Python 3.10 или новее.
- **Установка со всеми форматами:** `pip install 'markitdown[all]'`.
- **Точечная установка:** например, только PDF/DOCX/PPTX — `pip install 'markitdown[pdf, docx, pptx]'`.
- **Запуск:** `markitdown path-to-file.pdf > document.md` или с явным указанием файла на выходе — `markitdown path-to-file.pdf -o document.md`.

4. markitdown-mcp — автоконвертация в Claude Desktop

- Отдельный пакет `markitdown-mcp` поднимает MCP-сервер с одним инструментом — `convert_to_markdown(uri)`, принимающим ссылки `http:`, `https:`, `file:` или `data:`.
- **Установка:** `pip install markitdown-mcp`; запуск по STDIO — команда `markitdown-mcp`.
- Для Claude Desktop разработчики рекомендуют Docker-образ: собрать через `docker build -t markitdown-mcp:latest .`, затем в `claude_desktop_config.json` прописать сервер командой `docker run --rm -i markitdown-mcp:latest`.
- Сервер по умолчанию слушает только `localhost` — это осознанное ограничение безопасности, а не недоработка; не открывай его на другие интерфейсы без понимания последствий.

5. Плагины: расширение через сторонние конвертеры

- Плагин `markdown-ocr` добавляет OCR к конвертерам PDF, DOCX, PPTX и XLSX, вытаскивая текст со встроенных изображений через LLM-зрение — по той же схеме `llm_client / llm_model`, что уже используется для описания картинок.
- **Установка:** `pip install markdown-ocr` плюс любой OpenAI-совместимый клиент, например `pip install openai`.
- Сторонние плагины ищутся на GitHub по хэштегу `#markdown-plugin`; без указанного `llm_client` плагин просто пропускает OCR и работает штатный конвертер.