

Пример разработки плагина ссылки

Суть нашего плагина будет во вставке часто используемых шаблонов описания ссылок, например: *Ссылка на оригинальную статью*, *См. также* и *Читать далее....* Статья затронет добавление кнопки нашего плагина на панель инструментов редактора (toolbar), описание процесса разработки и закончим переводом нашего плагина на другие языки.

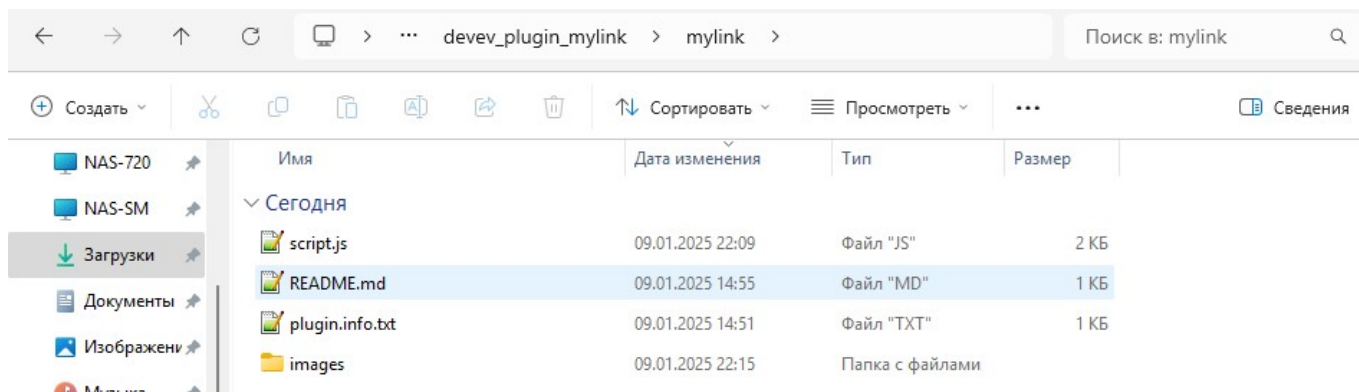
Подготовка папок и файлов

В папке `lib/plugins` создадим папку нашего плагина (например `link` или `mylink`). Я использую название `mylink`, т.к. более простые могут быть заняты. В нашей папке создадим файлы:

- `plugin.info.txt`
- `README.md`
- `script.js`

и папку:

- `images`



Файл `plugin.info.txt`

Каждый плагин должен предоставить некоторую базовую информацию о себе. Это делается через файл с именем, `plugin.info.txt` расположенный в каталоге плагина .

Вот пример того, как должен выглядеть такой файл.

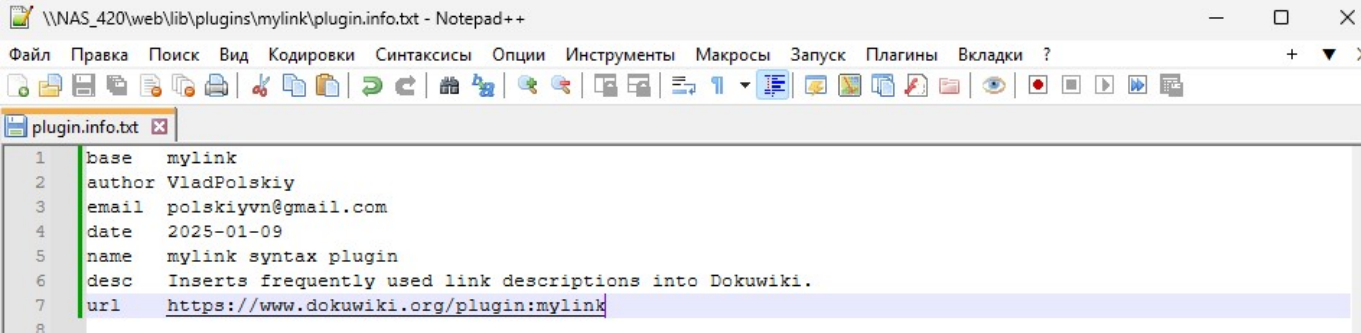
[plugin.info.txt](#)

```
base    mylink
author  VladPolSKIY
email   polskiyvn@gmail.com
date    2025-01-09
name    mylink syntax plugin
desc    Inserts frequently used link descriptions into Dokuwiki.
```

url <https://www.dokuwiki.org/plugin:mylink>

Параметр	Описание
base	Техническое имя плагина. Менеджер расширений установит его в этот каталог.
author	Полное имя автора плагина
email	Электронная почта для связи с автором плагина по поводу этого плагина
date	Дата последнего обновления этого плагина в формате ГГГГ-ММ-ДД. Не забудьте обновить это, когда будете обновлять свой плагин! ¹⁾
name	Удобочитаемое имя плагина
desc	Описание того, что делает плагин
url	URL-адрес , по которому можно получить более подробную информацию о плагине
minphp	Минимальная версия PHP, требуемая для этого плагина. Пусто для того же, что и DokuWiki. Только Major.Minor версия, например 8.2, не 8.2.3 может быть указана!
maxphp	Максимальная версия PHP, поддерживаемая этим плагином. Пусто для того же, что и DokuWiki. Только Major.Minor версия, например 8.2, не 8.2.3 может быть указана!

Примечание: plugin.info.txt заменяет старый вызов getInfo() в ваших компонентах плагина для DokuWiki 2009-12-25 «Lemming» и более поздних версий. getInfo() теперь реализован в базовом классе и будет считывать всю информацию из этого файла. Его больше не нужно реализовывать в ваших подклассах.



Файл README.md

README.md служит первым документом для пользователей и разработчиков, которые хотят понять назначение, настройку и использование проекта.

README.md

```
# плагин моих ссылок

Это плагин для [dokuwiki](https://www.dokuwiki.org/start) . .

См. [домашнюю страницу](https://www.dokuwiki.org/plugin:mylink) для
получения подробной документации. Здесь мы приводим краткое описание требуемого
синтаксиса:
```

```

` ``
{magnifier-lefticon}См. так-же [[:example|]]
{blue-document--arrowicon}[[http://example.com| Ссылка на оригинальную
статью]]
` ``

```

- Квадратные скобки обозначают ссылку.

Файл script.js

JavaScript файл с рабочим кодом нашего плагина (подключения кнопки и вывод описания нашей ссылки в тело редактируемой страницы. Поскольку пользовательские и плагиновые скрипты загружаются при всех запросах, а панель инструментов инициализируется только при редактировании, нам нужно сначала убедиться, что массив панелей инструментов существует. Проверим это, `window.toolbar` определен ли он. Если определен, вы можете добавить свою кнопку в конец массива.

[script.js](#)

```

/* JavaScript function to create mylink toolbar in Dokuwiki */
/* see http://www.dokuwiki.org/plugin:mylink for more info */

/* Путь к папке с иконками для кнопок панели */
mylink_icobase = "../..plugins/mylink/images/";

if(window.toolbar != undefined) {           // Проверка наличия массива панели
инструментов
    toolbar[toolbar.length] = {             // Кнопка верхнего уровня панели
инструментов
        "type":"picker",                    // тип действия кнопки: picker⇒
меню выбора
        "title":"My links",                  // название кнопки, отображаемое
при наведении курсора
        "icon":mylink_icobase+"toolbar_icon.png", //значок, используемый для
кнопки
        "list":[{ //список выпадающего меню кнопок
            "type":"format", // 1я кнопка в выпадающем окне. тип действия кнопки:
format⇒ вики-разметка
            "title":"Link to the original article", // название кнопки,
отображаемое при наведении курсора
            "icon":mylink_icobase+"blog.png", //путь к иконке, используемой для
кнопки
            "open":"[[",
            "sample":"http://example.com|Link to the original article",
            "close":"]]"
        }, {
            "type":"format", // 2я кнопка в выпадающем окне. тип действия кнопки:

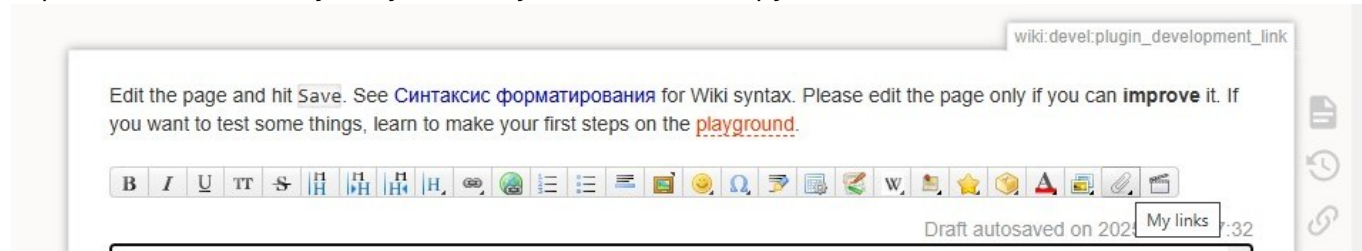
```

```
format⇒ вики-разметка
    "title":"See also",    // название кнопки, отображаемое при наведении
    курсора
    "icon":mylink_icobase+"magnifier-medium-left.png", //путь к иконке,
    используемой для кнопки
    "open":"See also: [[",  //открывающий тег вики-разметки для вставки
    "sample":"http://example.com|",
    "close":"]]"           //закрывающий тег вики-разметки для вставки
    }
    ]
};
}
```

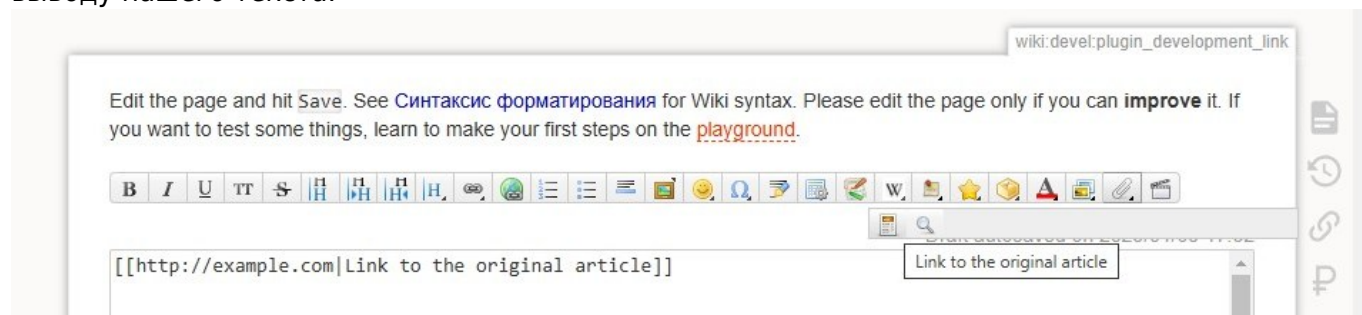
в данном случае у нас

```
    "open":"See also: [[",  //открывающий тег вики-разметки для вставки
    "sample":"http://example.com|",
    "close":"]]"           //закрывающий тег вики-разметки для вставки
представляют собою разрезанный код для вывода:
See also: [[ http://example.com| ]]
```

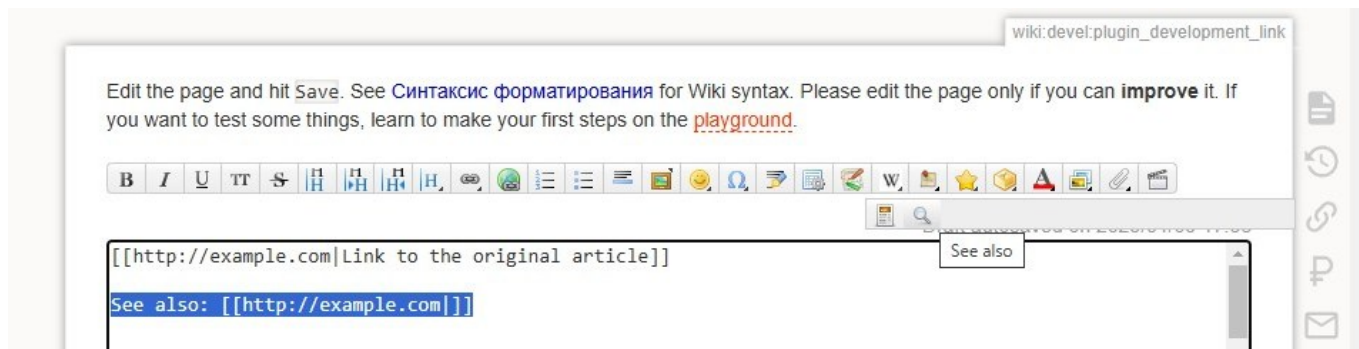
После редактирования и сохранения с нашим кодом файла script.js, переходим в редактор страниц и видим нашу новую кнопку на панели инструментовЖ



При нажатии на кнопку проверяем открытие выпадающего списка кнопок и работу кода по выводу нашего текста:



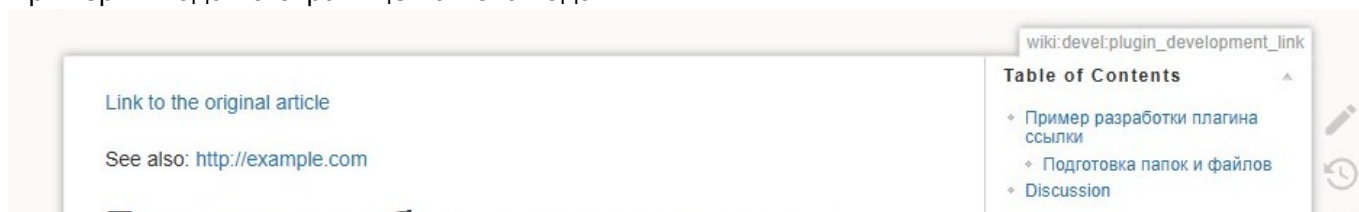
Проверяем вторую кнопку:



Мы видим код, который мы разрезали на части в файле script.js

```
See also: [[http://example.com|]]
```

Пример вывода на странице нашего кода:

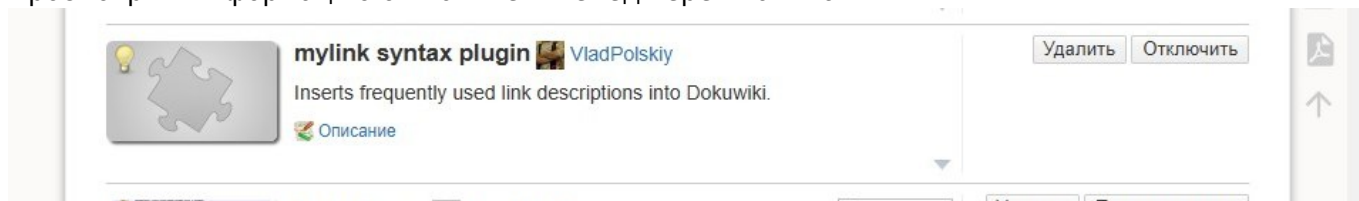


Ниже вы видите, как это отображается на этой странице, что говорит о том, что наш код работаетю [Link to the original article](#)

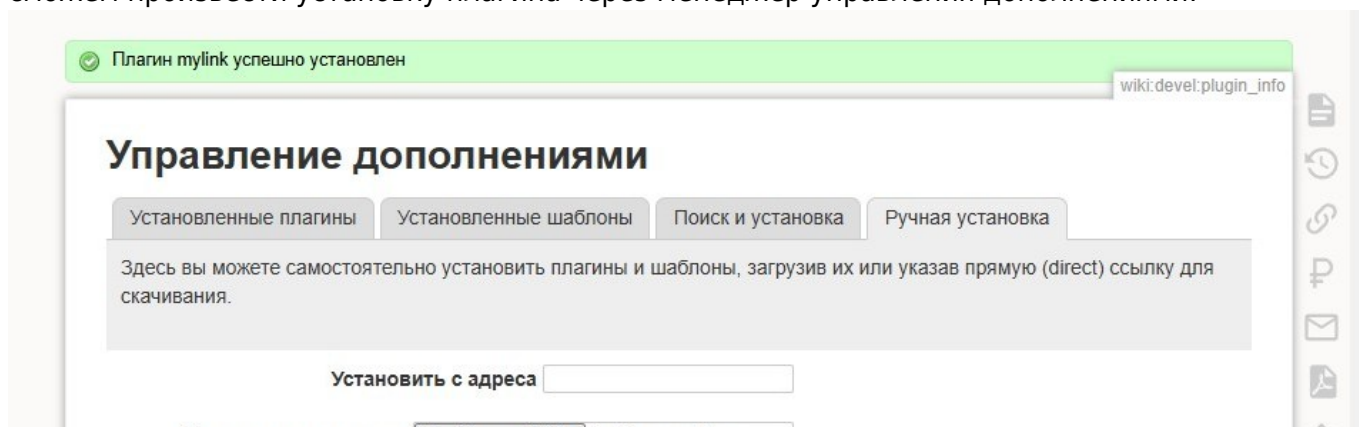
See also: <http://example.com>

Регистрация в программе нашего плагина прошла автоматически, т.к. мы создавали наш проект в папке lib/plugins. При этом мы можем заархивировать наш проект и установить через менеджер плагинов.

Посмотрим информацию о плагине в менеджере плагинов:



Если мы удалим наш проект из lib/plugins и заархивируем нашу папку в формате .zip, то сможем произвести установку плагина через менеджер управления дополнениями:



Перевод плагина на разные языки

1)

Эта дата должна быть равна дате «Последнее обновление» на странице плагинов в репозитории. Это необходимо для автоматической маркировки обновлений, используемой

Extension Manager, представленной в релизе Ponder Stibbons

From:

<https://wwoss.direct.quickconnect.to/> - worldwide open-source software

Permanent link:

https://wwoss.direct.quickconnect.to/doku.php?id=wiki:devel:plugin_development_link&rev=1736458735

Last update: **2025/01/10 00:38**

