

[/dev/notes](#)

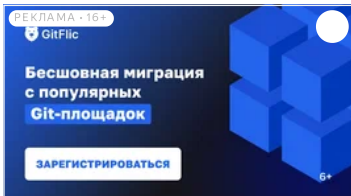
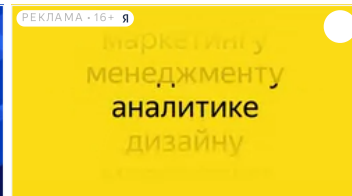

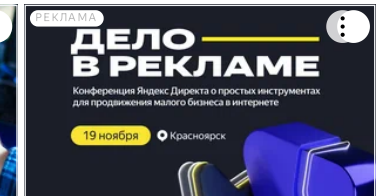
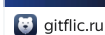



Опубликовано: суббота, 29 апреля 2023 г. в 08:54

# Vim: Установка плагинов

Vim

Источник: «Plugin installation»

Хотя мы склоняемся к нативной функциональности Vim в этом списке, большинство опытных пользователей Vim в какой-то момент расширяют возможности Vim с помощью любого из тысяч доступных плагинов с открытым исходным кодом. Если вы не живёте и не дышите Vim, вы можете не знать, сколько существует способов установки и управления плагинами, включая встроенный метод, поставляемый с Vim 8.

 <p>РЕКЛАМА -16+ GitFlic Бесплатная миграция с популярных Git-площадок ЗАРЕГИСТРИРОВАТЬСЯ</p>	 <p>РЕКЛАМА -16+ Я маркетингу менеджменту аналитике дизайну</p>	 <p>РЕКЛАМА -16+ skillbox.ru Научитесь программировать микроконтроллеры на языке C++ Узнать больше</p>	 <p>РЕКЛАМА ДЕЛО В РЕКЛАМЕ Конференция Яндекс Директа о простых инструментах для продвижения малого бизнеса в интернете 19 ноября • Красноярск</p>
 <p>gitflic.ru GitFlic - Российская платформа для разработчиков ПО Зарегистрироваться</p>	 <p>practicum.yandex.ru Начните изучать курс «Разработчик C++» бесплатно Узнать больше</p>	 <p>skillbox.ru Научитесь программировать микроконтроллеры на языке C++ Узнать больше</p>	 <p>yandex.ru Простые рекламные инструменты и практические советы Зарегистрироваться</p>

Если вы всё ещё используете один из неродных менеджеров пакетов, возможно вы упускаете один из лучших приёмов нативного менеджера пакетов Vim: Возможность загрузки плагинов на лету. Подробности будут дальше. Просим извинение, за отклонение от обычного формата кратких советов, но вот обзор четырёх различных способов установки плагинов Vim.

## Нативный менеджер пакетов Vim

Начиная с версии 8, Vim поставляется с нативной возможностью управления пакетами. Если вы используете Vim 8 и любой из приведённых далее менеджеров плагинов, в этом нет необходимости. (Если вы не используете Vim 8 или старше, почему бы и нет?) Чтобы установить плагины в Vim 8, они просто должны находиться в нужном месте, и Vim найдёт их и загрузит. Это место может быть где угодно в `~/.vim/pack/*/start`, так что вот как это настроить:

```
mkdir -p ~/.vim/pack/vendor/start
```

Мы будем использовать соглашение поставщиков, поскольку это библиотеки сторонних поставщиков, но вам это не обязательно. Вы можете создать отдельные каталоги для цветовых схем, синтаксиса и т.д. Затем любой плагин Vim можно просто поместить в любой из этих каталогов. Большинство копирует плагины через Git следующим образом:

```
git clone https://github.com/tpope/vim-vividchalk ~/.vim/pack/vendor/start/vim-vividchalk
```

Преимущество использования Git, конечно же, в простоте обновления. Вы можете выполнить `git pull` внутри любого из этих каталогов плагинов, чтобы обновить их. Или, как делаю, регулярно обновляю все сразу:

```
for i in ~/.vim/pack/vendor/start/*; do git -C $i pull; done
```

Одной из самых полезных функций родного/нативного менеджера пакетов Vim является возможность опциональной загрузки некоторых плагинов. Вместо каталога `start` вы можете использовать каталог `opt`, и эти плагины не загружаются автоматически. Вместо этого вы можете загружать их по запросу внутри Vim:

```
:packadd plugin-name
```

Вышеперечисленно также работает в NeoVim, но используется другое расположение каталога: `~/.local/share/nvim/site/pack`.

# Pathogen

**Pathogen** — является пионером менеджеров пакетов Vim, проложив путь для того, чтобы Vim принял аналогичную функциональность, а другие менеджеры пакетов предложили несколько иные методы установки. **Установить Pathogen** очень просто: просто скопируйте файл `pathogen.vim` в каталог `~/.vim/autoload` и добавьте следующее в ваш `.vimrc`:

```
execute pathogen#infect()
```

Теперь любые подключаемые модули, которые вы хотите установить, можно просто поместить в их собственные подкаталоги в `~/.vim/bundle`, и они будут автоматически загружены при запуске Vim. Если вы клонируете плагины через Git, вы можете использовать описанный выше метод для обновления всех плагинов одновременно.

Итак, почему вы хотите использовать Pathogen вместо нативных возможностей Vim? Ну, а может, и нет. Фактически Tim Pope, разработчик Pathogen, рекомендует использовать встроенное управление пакетами Vim.

## Vim-Plug

Vim-Plug претендует на звание «минималистического менеджера плагинов Vim», но, напротив, он обладает множеством функций и предлагает гораздо больше возможностей, чем родные пакеты и Pathogen. В частности, возможность обновлять плагины из Vim, включая просмотр и откат обновлений.

В README файле есть подробные инструкции по **установке** и использованию менеджера плагинов Vim-Plug. Примечательно, что вместо того, чтобы просто читать каталог каталогов, Vim-Plug требует указать плагины, которые будут установлены внутри вашего `.vimrc`. Например:

```
" the essential Git plugin
Plug 'tpope/vim-fugitive'
```

Формат — это привычные имя владельца GitHub и имя репозитория. Одно из преимуществ этого стиля установки плагинов является возможность аннотировать использование плагина комментариями. И есть множество опций, которые вы можете указать, например, загрузка определённой ветки или загрузка плагина по требованию в определённых типах файлов, поэтому посмотрите **примеры использования**.

После того как вы указали свои плагины, просто запустите `:PlugInstall`, для установки новых. Или, обновите все свои плагины с помощью `:PlugUpdate`.

## Vundle

Самый популярный менеджер пакетов по звёздам GitHub, **Vundle** похож на Vim-Plug тем, что позволяет обновлять плагины в Vim. Вы также указываете плагины для установки прямо внутри `.vimrc`, аналогично Vim-Plug.

**Инсталляция**, описанная в README, происходит легко и быстро. Формат ссылок на плагины идентичен Vim-Plug, но использует ключевое слово `Plugin` вместо `Plug`. Вот так:

```
" handles brackets, parens, quotes, etc.
Plugin 'tpope/vim-surround'
```

Для установки новых плагинов с помощью Vundle, запустите `:PluginInstall` или добавьте к нему восклицательный знак для обновления плагинов `:PluginInstall!` или, чтобы обновить установленные плагины можно воспользоваться командой `:PluginUpdate`

Предыдущая Статья

[Новое в Symfony 6.3 – Улучшения Form](#)

Следующая Статья

[В чём разница между composer.json и composer.lock](#)

Ссылки

[Главная](#)

[Статьи](#)

[Тэги](#)

[О Проекте](#)

Социальные медиа

[Дзен](#)

[Boosty](#)

[Telegram](#)

[VK](#)

[Twitter](#)

[Github](#)

Тэги

[BEM](#) [Backend](#) [CSS](#) [Composer](#) [Database](#) [DevOps](#) [Eloquent](#) [Frontend](#) [Git](#) [GraphQL](#) [Grid](#) [Guide](#) [Gulp](#) [HTML](#) [JAMStack](#) [JavaScript](#) [Laravel](#)  
[NodeJS](#) [PHP](#) [PostgreSQL](#) [Python](#) [React](#) [SCSS](#) [SEO](#) [SQL](#) [SSG](#) [Sass](#) [Security](#) [Symfony](#) [TDD](#) [Tailwind](#) [Testing](#) [Tools](#) [TypeScript](#) [Vim](#) [Vite](#)  
[a11y](#) [docker](#) [nginx](#)

Поддержите проект

Появится предупреждение, что вы переводите физлицу, в кошелёк 4100147881733.

Сумма:  ₽

- ☐ Из ЮMoney
- ☐ Банковской картой

[Поддержать](#)