# Машинный перевод статей jetbrains.com (<https://www.jetbrains.com/help/phpstorm/using-git-integration.html>)

Эта инструкция для версии PhpStorm 2020.3

# Настройка PhpStorm для использования GitHub

Установить пароли для пультов Git**﻿**

Git Settings: **Settings/Preferences | Version Control | Git**

Required plugin: **Git** (bundled and enabled by default)

Каждый раз, когда вы взаимодействуете с удаленным репозиторием Git (например, во время операции извлечения, обновления или отправки), требуется авторизация. Вы можете настроить PhpStorm на запоминание ваших паролей, чтобы вам не приходилось указывать свои учетные данные каждый раз, когда требуется авторизация.

### Настроить политику паролей﻿

В диалоговом окне «Настройки / Предпочтения» Ctrl+Alt+S выберите «Внешний вид и поведение» | Системные настройки | Пароли в левой колонке.

**Settings/Appearance and Behavior | System Settings | Passwords**.

Выберите, как вы хотите, чтобы PhpStorm обрабатывал пароли для удаленных репозиториев Git:

В KeePass: выберите этот вариант, чтобы использовать диспетчер паролей KeePass для хранения ваших паролей. Когда вы используете менеджер паролей KeePass , мастер-пароль будет использоваться для доступа к файлу, в котором хранятся индивидуальные пароли. Как только PhpStorm запомнит ваши пароли, он не будет запрашивать их, если вам не понадобится доступ к базе данных паролей. Введите пароль, который будет использоваться для доступа к файлу c.kdbx, в поле MasterPassword.

Вы можете изменить расположение файла c.kdbx по умолчанию в поле « База данных».

Чтобы импортировать файл c.kdbx , щелкните значок настроек и выберите «Импорт» в раскрывающемся меню или щелкните кнопку "Обзор» и укажите путь к локальному файлу, содержащему ваши пароли.

Если вы хотите удалить существующие пароли из базы данных, выберите Очистить.

Не сохранять, забывать пароли после перезапуска: выберите эту опцию, если вы хотите, чтобы ваши пароли сбрасывались после закрытия PhpStorm.

GitHub﻿

PhpStorm позволяет вам управлять проектами Git, размещенными на GitHub, непосредственно из IDE: клонировать репозитории, делиться своими проектами, создавать форки, делиться кодом через сущности, создавать запросы на вытягивание и просматривать входящие запросы на вытягивание.

### Зарегистрируйте учетную запись GitHub﻿

Чтобы иметь возможность получать данные из репозитория, размещенного на GitHub, или делиться своими проектами, вам необходимо зарегистрировать свою учетную запись GitHub в PhpStorm.

Если вы не хотите указывать свои учетные данные каждый раз при синхронизации с удаленным компьютером или подталкивать свои коммиты, вы можете настроить PhpStorm для сохранения информации вашей учетной записи (см. Раздел Настройка политики паролей).

#### Зарегистрируйте существующую учетную запись﻿

В диалоговом окне « Настройки / Предпочтения » Ctrl+Alt+S выберите « Контроль версий | GitHub на левой панели.

Щелкните кнопка Добавить.

В открывшемся диалоговом окне укажите URL-адрес вашего сервера GitHub (либо github.com, либо корпоративный экземпляр).

Выполните одно из следующих действий:

Если у вас уже есть токен, щелкните ссылку Использовать токен и вставьте его туда.

Если вы хотите получить новый токен, введите свой логин и пароль. Если у вас включена двухфакторная аутентификация, вам будет предложено ввести код, который будет отправлен вам по SMS или через мобильное приложение. Дополнительные сведения о токенах GitHub см. В разделе « Создание токена личного доступа».

Для токена должны быть включены области репозитория, сути и read: org в разрешениях вашей учетной записи (см. Общие сведения об областях).

Создайте новую учетную запись GitHub﻿

В диалоговом окне « Настройки / Предпочтения » Ctrl+Alt+S выберите « Контроль версий | GitHub на левой панели.

Нажмите Добавить учетную запись и в открывшемся диалоговом окне нажмите Зарегистрироваться в Github .

Зарегистрируйте свою учетную запись на открывшейся странице регистрации на GitHub .

Вернитесь к настройкам PhpStorm и укажите свои учетные данные.

Управляйте несколькими учетными записями﻿

Вы можете использовать несколько учетных записей GitHub в PhpStorm: например, личную учетную запись для работы над проектом с открытым исходным кодом и корпоративную учетную запись для вашей основной работы.

В диалоговом окне « Настройки / Предпочтения » Ctrl+Alt+S выберите « Контроль версий | GitHub на левой панели.

Используйте добавить значок-кнопку, чтобы добавить столько учетных записей, сколько вам нужно.

(Необязательно) Чтобы установить учетную запись по умолчанию для текущего проекта, выберите ее и щелкните кнопка Проверить. Если установлена ​​учетная запись по умолчанию, PhpStorm не попросит вас выбрать учетную запись, которую вы хотите использовать, когда вы поделитесь своим проектом на GitHub , перебазируете вилку, создадите запрос на перенос или создадите суть.

Управляйте проектами, размещенными на GitHub﻿

### Проверить проект (клон)﻿

Вы можете клонировать репозиторий, в который хотите внести свой вклад, прямо из PhpStorm и создать на его основе новый проект.

В главном меню выберите Git | Клонировать.

В диалоговом окне « Получить из системы контроля версий» слева выберите GitHub .

Укажите URL-адрес репозитория, который вы хотите клонировать. Вы можете выбрать репозиторий из списка всех проектов GitHub, связанных с вашей учетной записью и организацией, которой принадлежит ваша учетная запись.

В поле Каталог введите путь к папке, в которой будет создан ваш локальный репозиторий Git.

Щелкните Клонировать. Если вы хотите создать проект на основе этих источников, нажмите Да в диалоговом окне подтверждения. PhpStorm автоматически установит корневое сопоставление Git с корневым каталогом проекта.

### Поделиться проектом на GitHub﻿

Вы можете добавить удаленный репозиторий GitHub для проекта, который вы разрабатываете локально, чтобы другие могли просматривать его или вносить в него свой вклад.

Откройте проект, которым хотите поделиться.

В главном меню выберите Git | GitHub | Поделитесь проектом на GitHub .

Если вы уже зарегистрировали свою учетную запись GitHub в PhpStorm, соединение будет установлено с использованием этих учетных данных.

Если вы не зарегистрировали свою учетную запись в PhpStorm, откроется диалоговое окно «Вход в GitHub». Укажите свой токен доступа или запросите новый с вашим логином и паролем.

Когда соединение с GitHub будет установлено, откроется диалоговое окно «Поделиться проектом в GitHub». Укажите имя нового репозитория, имя пульта дистанционного управления и введите описание вашего проекта.

Вы можете выбрать опцию Private, если не хотите разрешать публичный доступ к вашему репозиторию другим пользователям GitHub.

Нажмите « Поделиться», чтобы создать новый репозиторий и загрузить в него исходники проекта.

Перейти к версии файла на GitHub﻿

Вы можете перейти от PhpStorm к версии файла на GitHub. PhpStorm определяет, какая ветка активна в данный момент, а также последняя версия файла, и открывает копию GitHub выбранного файла в соответствующем коммите в вашем веб-браузере. Если вы открываете версию файла GitHub из редактора, файл также будет автоматически прокручен до текущей строки.

Выберите файл в редакторе или в представлении «Проект» и выберите « Открыть на GitHub» в контекстном меню или во Ctrl+Shift+A в сплывающем окне. Удаленная версия файла откроется в браузере.

Если файл указывает на более чем один удаленный компьютер, вам будет предложено выбрать соответствующий репозиторий.

**Настройка npm для использования с пакетами GitHub**

Можно настроить npm для публикации пакетов в пакетах GitHub и использования пакетов, хранящихся в пакетах GitHub, в качестве зависимостей в проекте npm.

Пакеты гитхаб доступен бесплатно на GitHub и GitHub про команды на GitHub, GitHub и облачных предприятия, и GitHub один. Пакеты GitHub недоступны для частных репозиториев, принадлежащих учетным записям, использующим устаревшие планы для каждого репозитория. Дополнительные сведения см. В разделе " Продукты GitHub."Аутентификация в пакетах GitHub

Маркер доступа необходим для публикации, установки и удаления пакетов в пакетах GitHub. Вы можете использовать персональный маркер доступа для аутентификации с вашим именем пользователя непосредственно в пакетах GitHub или API GitHub. Вы можете использовать GITHUB\_TOKEN для аутентификации с помощью рабочего процесса действий GitHub.

Аутентификация с помощью персонального маркера доступа

Для публикации и установки пакетов в пакетах GitHub необходимо использовать персональный маркер доступа с соответствующими областями действия. Дополнительные сведения см. В разделе " о пакетах GitHub."

Вы можете проверить подлинность на GitHub пакеты с НПМ, либо редактирования каждого пользователя ~/.файл npmrc для включения вашего персонального маркера доступа или путем входа в npm в командной строке с использованием вашего имени пользователя и персонального маркера доступа.

Аутентификация путем добавления вашего персонального маркера доступа к вашему ~/.файл npmrc, отредактируйте ~/.файл npmrc для вашего проекта должен содержать следующую строку, заменяющую токен на ваш личный токен доступа. Создайте новый ~/.файл npmrc, если он не существует.

//npm.pkg.github.com/:\_authToken=TOKEN

Для аутентификации при входе в npm используйте команду npm login, заменив имя пользователя на ваше имя пользователя GitHub, токен на ваш личный токен доступа и публичный адрес электронной почты на ваш адрес электронной почты.

$ НПМ логин-реестра=на HTTPS://НПМ.ПКГ.на GitHub.ком

> Имя пользователя: USERNAME

> Пароль: токен

> Электронная почта: публичный адрес электронной почты

Аутентификация с помощью GITHUB\_TOKEN

Если вы используете рабочий процесс действий GitHub, вы можете использовать GITHUB\_TOKEN для публикации и использования пакетов в пакетах GitHub без необходимости хранить и управлять персональным маркером доступа. Дополнительные сведения см. В разделе " аутентификация с помощью GITHUB\_TOKEN."

Публикация пакета

По умолчанию пакеты GitHub публикуют пакет в репозитории GitHub, указанном в поле имя пакета.файл json. Например, можно опубликовать пакет с именем @мой орг/тест на моей орг/тестовый репозиторий GitHub. Вы можете добавить сводку для страницы списка пакетов, включив README.md файл в каталоге пакета. Дополнительные сведения см. В разделе " Работа с пакетом.json " и "как создать узел.модули js " в документации npm.

Вы можете опубликовать несколько пакетов в одном репозитории GitHub, включив в пакет поле URL.файл json. Дополнительные сведения см. В разделе " публикация нескольких пакетов в одном репозитории."

Вы можете настроить отображение области для вашего проекта, используя либо локальный .файл npmrc в проекте или с помощью опции publishConfig в пакете.формат JSON. Пакеты GitHub поддерживают только пакеты npm с областью действия. Пакеты с областью действия имеют имена в формате @owner / name. Пакеты с областью действия всегда начинаются с символа@. Возможно, Вам потребуется обновить имя в вашем пакете.json, чтобы использовать имя области. Например, "имя":"@codertocat/hello-world-npm".

После публикации пакета его можно просмотреть на GitHub. Дополнительные сведения см. В разделе " Просмотр пакетов."

Публикация пакета с использованием локального сервера .файл npmrc

Вы можете использовать .npmrc файл, чтобы настроить отображение рамки для вашего проекта. В.файл npmrc, используйте URL пакетов GitHub и владельца учетной записи, чтобы пакеты GitHub знали, куда направлять запросы пакетов. С помощью .файл npmrc предотвращает случайную публикацию пакета другими разработчиками в npmjs.org вместо пакетов GitHub. Поскольку прописные буквы не поддерживаются, вы должны использовать строчные буквы для владельца репозитория, даже если имя пользователя или организации GitHub содержит прописные буквы.

Аутентификация в пакетах GitHub. Дополнительные сведения см. В разделе " аутентификация в пакетах GitHub."

В том же каталоге, что и ваш пакет.файл json, создание или редактирование.файл npmrc должен содержать строку с указанием URL-адреса пакетов GitHub и владельца учетной записи. Замените имя владельца именем пользователя или учетной записи организации, которой принадлежит хранилище, содержащее ваш проект.

реестра=на HTTPS://НПМ.ПКГ.на GitHub.ком/владелец

Добавьте свое .файл npmrc в репозиторий, где пакеты GitHub могут найти ваш проект. Дополнительные сведения см. В разделе " добавление файла в репозиторий с помощью командной строки."

Проверьте имя вашего пакета в пакете вашего проекта.формат JSON. Поле name должно содержать область действия и имя пакета. Например, если ваш пакет называется "test", и вы публикуете его в организации GitHub" My-org", поле name в вашем пакете.json должен быть @my-org / test.

Проверьте поле репозитория в пакете вашего проекта.формат JSON. Поле хранилище должно совпадать с URL-адрес репозитория GitHub. Например, если URL-адрес репозитория github.com/my-org/test тогда поле репозитория должно быть git:/ / github. com / my-org / test.git.

Опубликовать пакет:

$ НПМ опубликовать

Публикация пакета с помощью publishConfig в пакете.файл json

В пакете можно использовать элемент publishConfig.JSON-файл для указания реестра, в котором должен быть опубликован пакет. Дополнительные сведения см. В разделе" publishConfig " документации по npm.

Отредактируйте пакет.файл json для вашего пакета и включите запись publishConfig.

"publishConfig": {

"реестр":"https://npm.pkg.github.com/"

},

Проверьте поле репозитория в пакете вашего проекта.формат JSON. Поле хранилище должно совпадать с URL-адрес репозитория GitHub. Например, если URL-адрес репозитория github.com/my-org/test тогда поле репозитория должно быть git:/ / github. com / my-org / test.git.

Опубликовать пакет:

$ НПМ опубликовать

Публикация нескольких пакетов в одном репозитории

Чтобы опубликовать несколько пакетов в одном репозитории, можно включить URL-адрес репозитория GitHub в поле репозитория пакета.файл json для каждого пакета. Пакеты GitHub будут соответствовать репозиторию на основе этого поля, а не на основе имени пакета. Если вы храните пакет.файл json вне корневого каталога вашего репозитория, вы можете использовать поле Каталог, чтобы указать место, где пакеты GitHub могут найти пакет.файлы json.

"хранилище" : {

"тип" : "git",

"адрес": "ssh://git@github.com/OWNER/REPOSITORY.git",

"каталог": "пакеты / имя"

},

Установка пакета

Вы можете установить пакеты из пакетов GitHub, добавив пакеты в качестве зависимостей в пакет.файл json для вашего проекта. Для получения дополнительной информации об использовании пакета.json в вашем проекте см. раздел " Работа с пакетом.json " в документации npm.

По умолчанию можно добавлять пакеты из одной организации. Дополнительные сведения см. В разделе установка пакетов из других организаций

Вы также должны добавить .файл npmrc в ваш проект, поэтому все запросы на установку пакетов будут проходить через пакеты GitHub. При маршрутизации всех запросов пакетов через пакеты GitHub можно использовать пакеты как с областью действия, так и без нее. npmjs.com дополнительные сведения см. В разделе" npm-scope " в документации по npm.

Аутентификация в пакетах GitHub. Дополнительные сведения см. В разделе " аутентификация в пакетах GitHub."

В том же каталоге, что и ваш пакет.файл json, создание или редактирование.файл npmrc должен содержать строку с указанием URL-адреса пакетов GitHub и владельца учетной записи. Замените имя владельца именем пользователя или учетной записи организации, которой принадлежит репозиторий, содержащий ваш проект.

реестра=на HTTPS://НПМ.ПКГ.на GitHub.ком/владелец

Добавьте свое .файл npmrc в репозиторий, где пакеты GitHub могут найти ваш проект. Дополнительные сведения см. В разделе " добавление файла в репозиторий с помощью командной строки."

Настройка пакета.JSON в ваш проект, чтобы использовать пакет, который вы устанавливаете. Чтобы добавить в пакет зависимости от вашего пакета.в файле json для пакетов GitHub укажите полное имя пакета, например @my-org / server. Для пакетов от npmjs.com, укажите полное имя, например @babel/core или @lodash. Например, это следующий пакет.в JSON используются @окто-орг/окто-пакета приложения в качестве зависимостей. {

"имя": "@my-org / server",

"версия": "1.0.0",

"описание": "приложение сервера, который использует @окто-орг/окто-пакета приложения",

"основной индекс.JS",

"автор": "",

"лицензия": "MIT",

"зависимости": {

"@окто-орг/окто-приложение": "1.0.0"

}

}

Установка пакета.

$ НПМ установить

Установка пакетов из других организаций

По умолчанию можно использовать только пакеты GitHub packages из одной организации. Если вы хотите направить запросы на пакет нескольким организациям и пользователям, вы можете добавить дополнительные строки .файл npmrc, заменяющий владельца именем пользователя или учетной записи организации, которой принадлежит репозиторий, содержащий ваш проект. Поскольку прописные буквы не поддерживаются, вы должны использовать строчные буквы для владельца репозитория, даже если имя пользователя или организации GitHub содержит прописные буквы.

реестра=на HTTPS://НПМ.ПКГ.на GitHub.ком/владелец

@Владелец:реестра=на HTTPS://НПМ.ПКГ.на GitHub.ком

@Владелец:реестра=на HTTPS://НПМ.ПКГ.на GitHub.ком