Оглавление

[Установка 2](#_Toc520478686)

[Общие настройки 2](#_Toc520478687)

[Интеграция с GitHub 2](#_Toc520478688)

[Настройки почты 2](#_Toc520478689)

[Создание и настройка заданий 3](#_Toc520478690)

[1. Создать новое задание 3](#_Toc520478691)

[2. Указать систему контроля версиями 3](#_Toc520478692)

[3. Настроить триггеры сборки 4](#_Toc520478693)

[4. Настройка среды сборки 4](#_Toc520478694)

[5. Заполнение этапов сборки 5](#_Toc520478695)

[6. Послесборочные операции 5](#_Toc520478696)

[Примеры скриптов 6](#_Toc520478697)

[Общие подходы 6](#_Toc520478698)

[Автоматическая сборка приложения 1С после изменения в Git 6](#_Toc520478699)

[Автоматическая проверка кода 1С после изменения в Git 6](#_Toc520478700)

# Установка

# Общие настройки

## Интеграция с GitHub

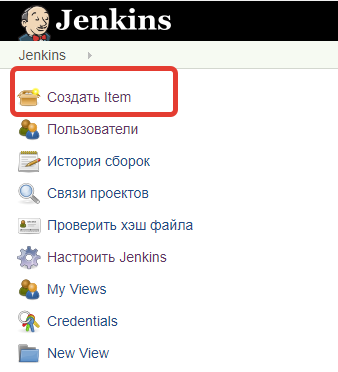
## Настройки почты

# Создание и настройка заданий

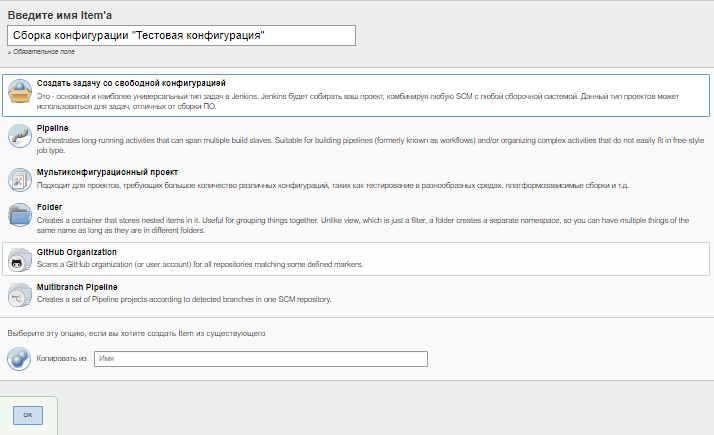
После внесения изменений в репозитории Git, сервер Jenkins запускает сборку готового приложения.

## Создать новое задание

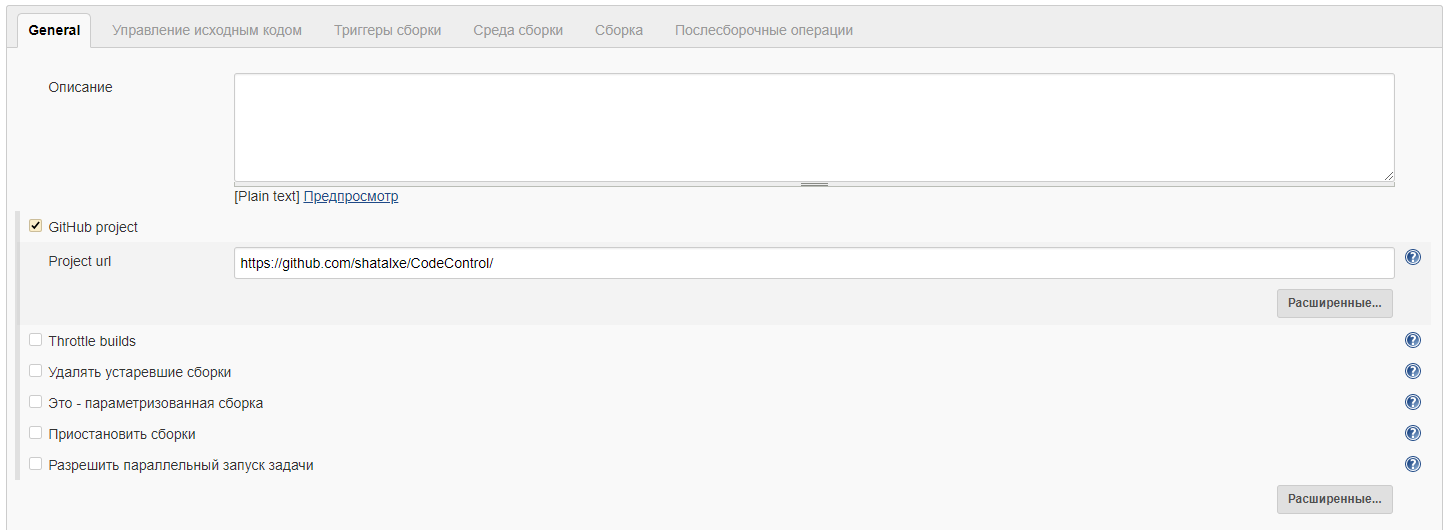
* 1. Создать Item



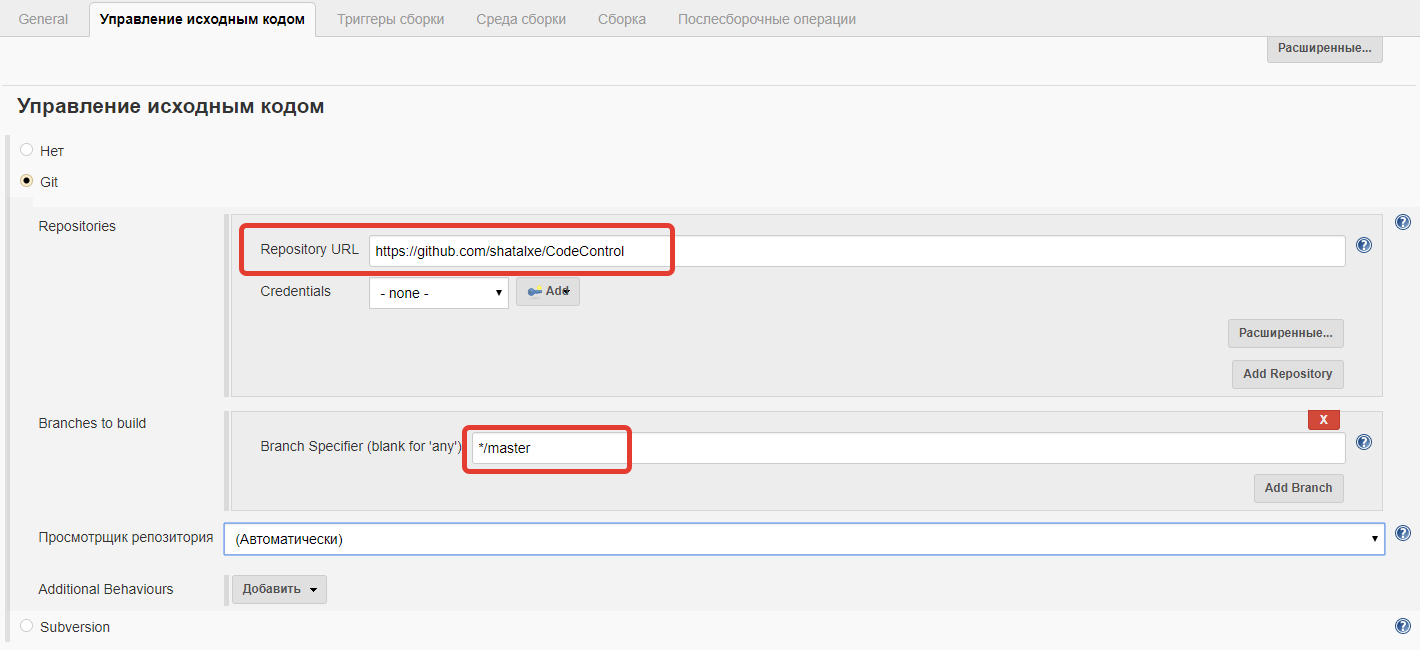
* 1. Выбрать **создать задачу со свободной конфигурацией**, задать имя, **Ok**



Откроется окно настройки задания



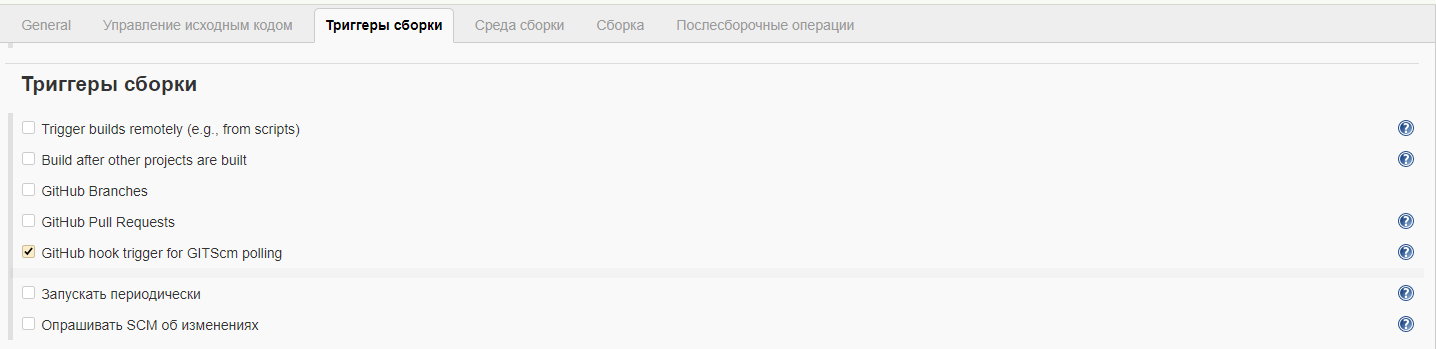
## Указать систему контроля версиями



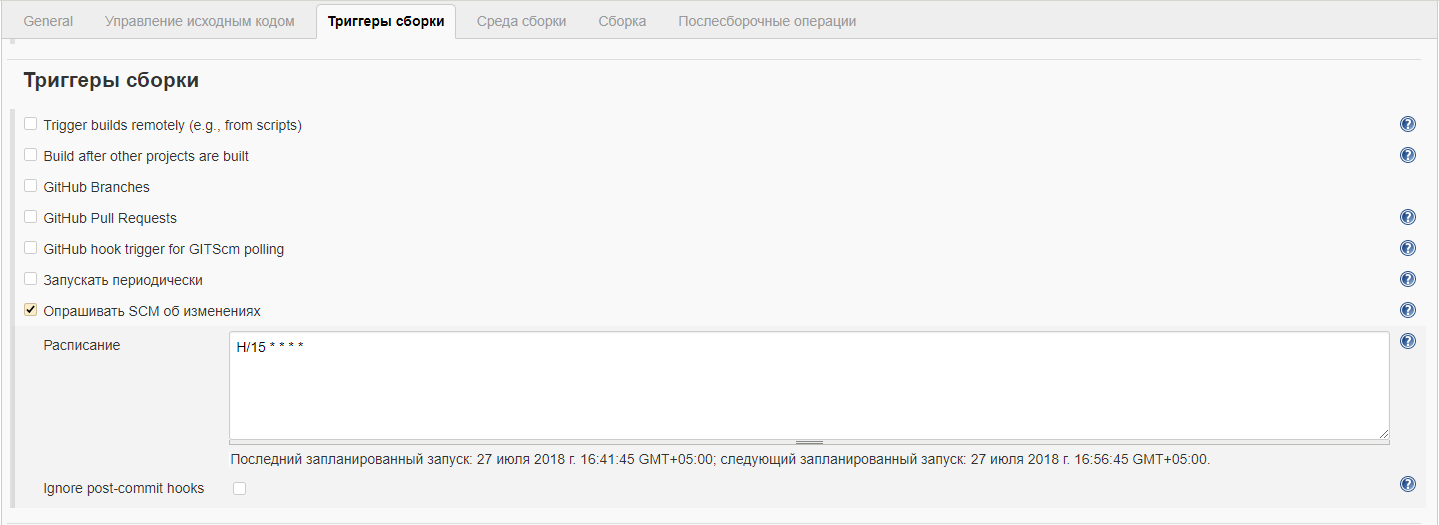
Для срабатывания задания по коммиту в любую ветку, в **поле Branch Specifier** поставить \*

## Настроить триггеры сборки

* Срабатывание по оповещению GitHub’а



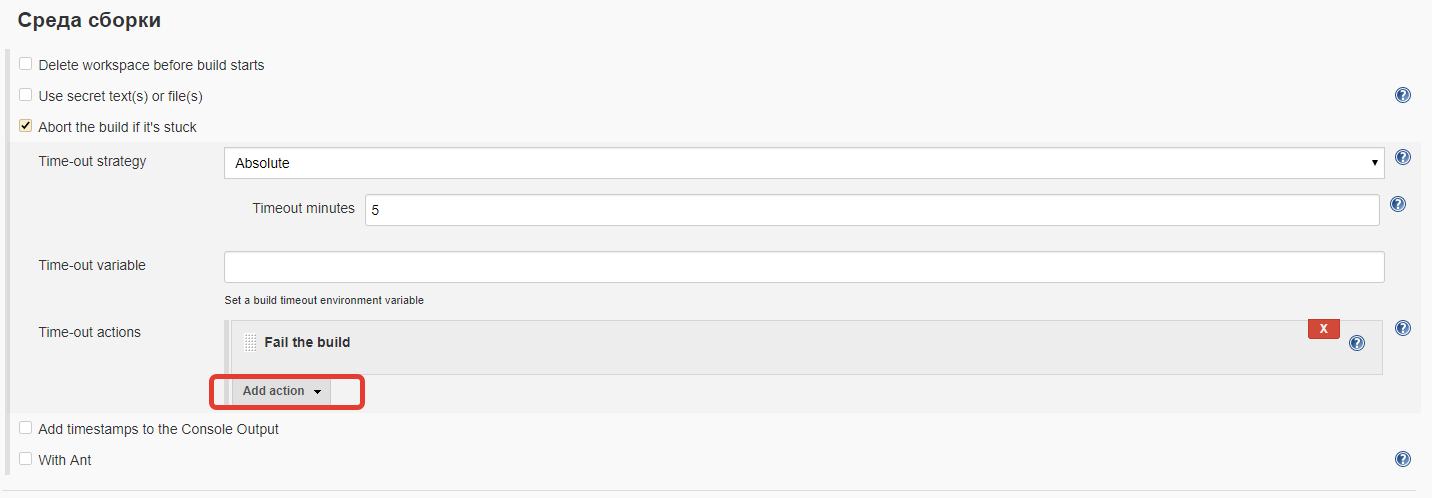
* Опрос Github по расписанию



## Настройка среды сборки

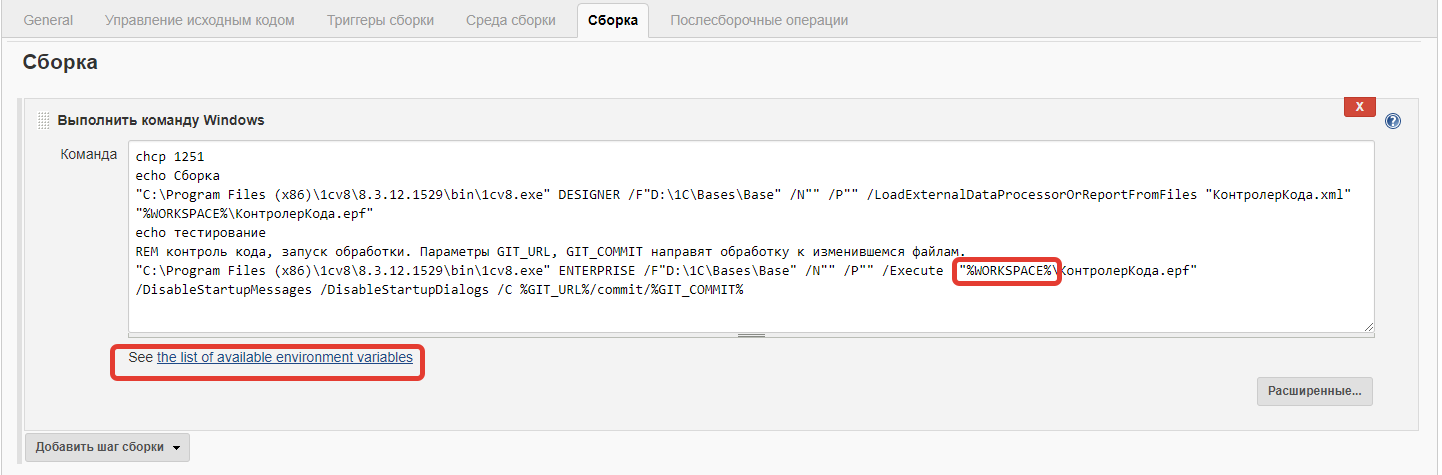
Для установки прекращения выполнения задания по таймауту, выставить флаг **Abort the build if it's stuck.**

Для того, чтобы сборка считалась неуспешной (например, чтобы разослать уведомления) нужно добавить событие (Add action) **Fail to build.**



## Заполнение этапов сборки

**Выполнить команду Windows**



Во время выполнения задания Jenkins по тексту скрипта соберет и выполнит bat-файл из **рабочего каталога** (WORKSPACE).

*Расположение* *bat-файла C:\Windows\TEMP\jenkins<id>.bat*

*Расположение* *рабочего каталога C:\Program Files (x86)\Jenkins\workspace\<имя проекта>\*

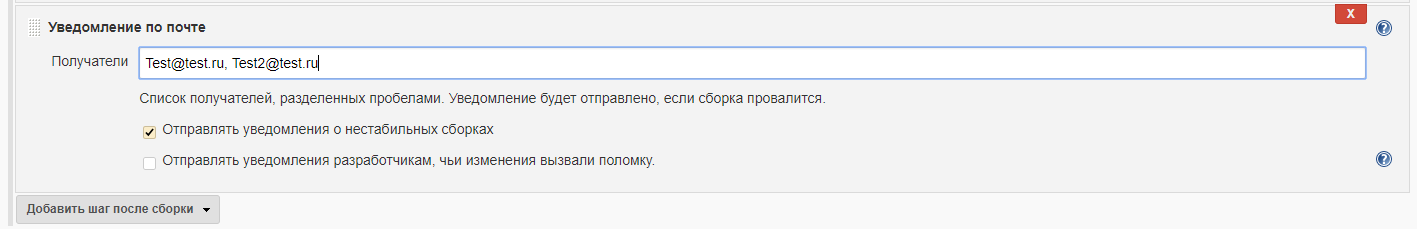
Допускается использование переменных. Список переменных **See the list of available environment variables.**

## Послесборочные операции

**Уведомление по почте**

Cборка будет считаться провалившейся, если сценарий выставит ненулевой %ERRORLEVEL%.

Указанному списку получателей будет отправлен лог выполнения.



**Editable Email Notification**

# Примеры скриптов

## Общие подходы

chcp 1251 – установить кодировку Windows-1251

В качестве пути к выгруженной конфигурации (обработке/отчету) достаточно указать текущий каталог, т.к. команда выполняется в рабочем каталоге (WORKSPACE), куда загружаются файлы из GitHub

Каталог сохранения должен быть доступен пользователю, от которого запущена служба Jenkins

Сложные конструкции следует комментировать (REM).

Параметры, заключенные в %% при создании bat-файла, будут заменены значением.

## Автоматическая сборка приложения 1С после изменения в Git

chcp 1251

"C:\Program Files (x86)\1cv8\8.3.12.1529\bin\1cv8.exe" DESIGNER /F"D:\1С\Bases\Base" /N"" /P"" /LoadExternalDataProcessorOrReportFromFiles "КонтролерКода.xml" "D:\КонтролерКода.epf"

## Автоматическая проверка кода 1С после изменения в Git

chcp 1251

"C:\Program Files (x86)\1cv8\8.3.12.1529\bin\1cv8.exe" DESIGNER /F"D:\1С\Bases\Base" /N"" /P"" /LoadExternalDataProcessorOrReportFromFiles "КонтролерКода.xml" "%WORKSPACE%\КонтролерКода.epf"

REM контроль кода, запуск обработки. Параметры GIT\_URL, GIT\_COMMIT направят обработку к изменившемся файлам.

"C:\Program Files (x86)\1cv8\8.3.12.1529\bin\1cv8.exe" ENTERPRISE /F"D:\1С\Bases\Base" /N"" /P"" /Execute "%WORKSPACE%\КонтролерКода.epf" /DisableStartupMessages /DisableStartupDialogs /C %GIT\_URL%/commit/%GIT\_COMMIT%