

Edit Windows Host to any site you want for your localhost IP 127.0.0.1

```
hosts - Notepad
File Edit Format View Help
# Copyright (c) 1993-2001 Microsoft Corp.
#
# This file has been automatically generated for use by Microsoft Internet
# Connection Sharing. It contains the mappings of IP addresses to host names
# for the home network. Please do not make changes to the HOSTS.ICS file.
# Any changes may result in a loss of connectivity between machines on the
# local network.
#

172.22.48.1 DESKTOP-JLEGGJC.mshome.net # 2028 6 6 10 15 17 59 882

2.168.137.51 rac1.mshome.net # 2023 2 0 26 22 47 38 81
1
#Oracle 19c RAC Virtualbox
192.168.24.41 rac-scan
192.168.24.42 rac-scan
192.168.24.43 rac-scan
127.0.0.1 ronjenkins.com
```

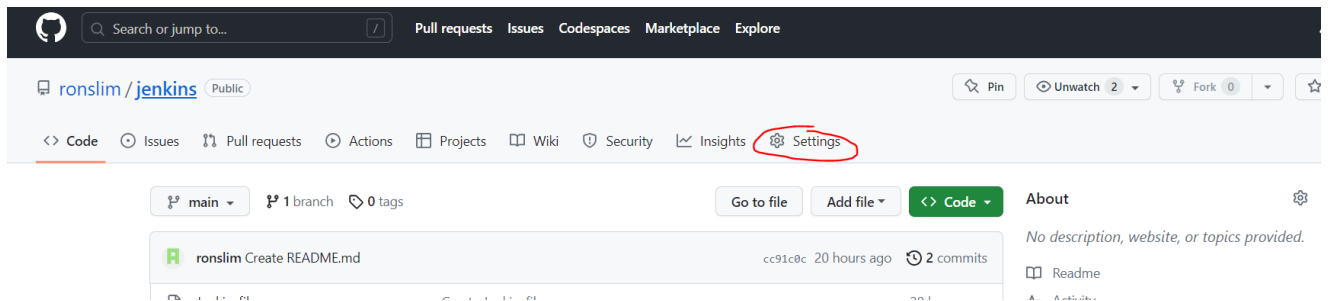
Execute jenkins.war via java to start Jenkins

```
Command Prompt - java -jar jenkins.war
Microsoft Windows [Version 10.0.19045.2965]
(c) Microsoft Corporation. All rights reserved.

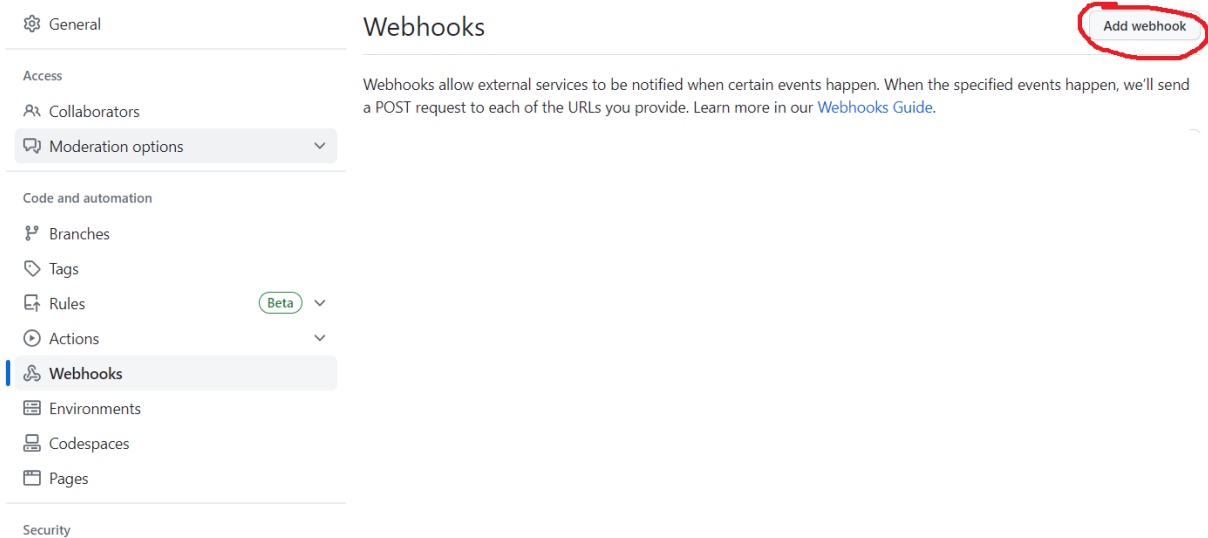
C:\Users\User>cd D:\Jenkins

D:\Jenkins>java -jar jenkins.war
Running from: D:\Jenkins\jenkins.war
webroot: D:\Jenkins\.jenkins\war
2023-06-12 20:01:46.503+0000 [id=1] INFO winstone.Logger#logInternal: Beginning extraction from war file
2023-06-12 20:01:46.558+0000 [id=1] WARNING o.e.j.s.handler.ContextHandler#setContextPath: Empty contextPath
2023-06-12 20:01:46.621+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: jetty-10.0.13; built: 2022-12-0
7T20:13:20.134Z; git: 1c2636ea05c0ca8de1ffd6ca7f3a98ac084c766d; jvm 17.0.7+8-LTS-224
2023-06-12 20:01:51.410+0000 [id=1] INFO o.e.j.w.StandardDescriptorProcessor#visitServlet: NO JSP Support for /,
did not find org.eclipse.jetty.jsp.JettyJspServlet
2023-06-12 20:01:51.461+0000 [id=1] INFO o.e.j.s.s.DefaultSessionIdManager#doStart: Session workerName=node0
2023-06-12 20:01:51.937+0000 [id=1] INFO hudson.WebAppMain#contextInitialized: Jenkins home directory: D:\Jenkins
\.jenkins found at: EnvVars.masterEnvVars.get("JENKINS_HOME")
2023-06-12 20:01:52.059+0000 [id=1] INFO o.e.j.s.handler.ContextHandler#doStart: Started w.@77b21474{Jenkins v2.4
01.1/,file:///D:/Jenkins/.jenkins/war/,AVAILABLE}{D:\Jenkins\.jenkins\war}
2023-06-12 20:01:52.082+0000 [id=1] INFO o.e.j.server.AbstractConnector#doStart: Started ServerConnector@62e136d3
{HTTP/1.1, (http/1.1)}{0.0.0.0:8080}
2023-06-12 20:01:52.096+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: Started Server@48524010{STARTIN
G}[10.0.13,sto=0] @6058ms
2023-06-12 20:01:52.097+0000 [id=28] INFO winstone.Logger#logInternal: Winstone Servlet Engine running: controlPor
t disabled
```

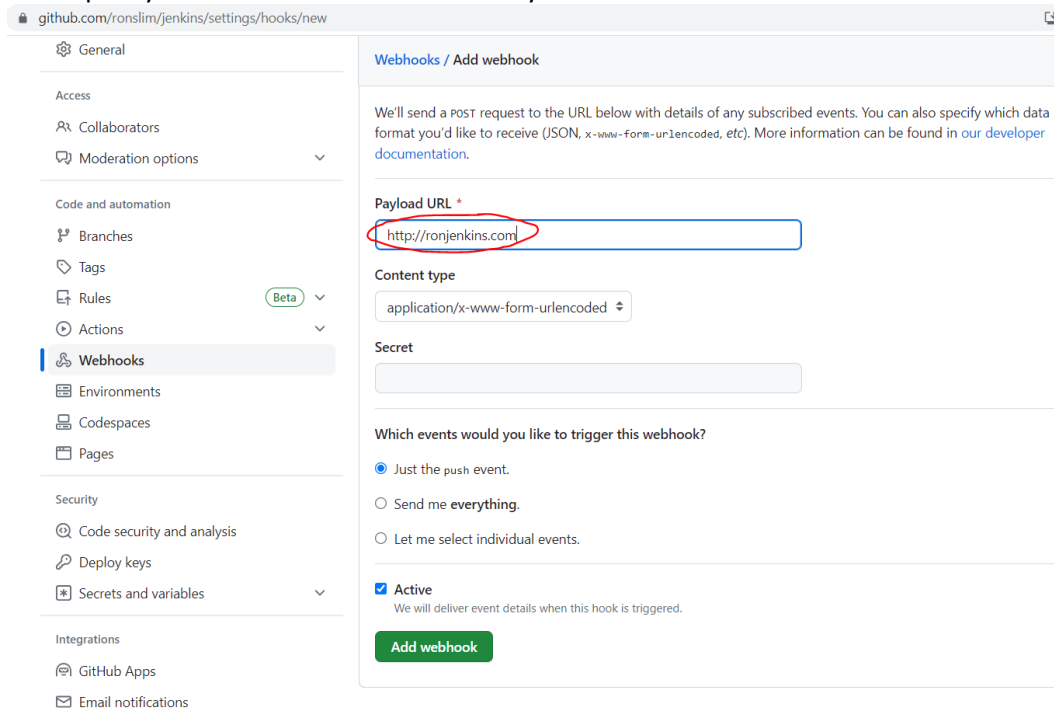
Go to github then to your Jenkins directory and choose settings tab



Go to webhooks and click Add Webhook



Then specify the site in Windows Host to Payload URL and then click Add Webhook



After that, check that the webhook is checked green. Green means successful communication but if red, there is a problem with site communication.

The screenshot shows the 'Webhooks' settings page in GitHub. On the left is a sidebar with navigation options: General, Access, Collaborators, Moderation options, Code and automation, Branches, Tags, Rules (marked Beta), Actions, Webhooks (selected), Environments, Codespaces, and Pages. The main content area is titled 'Webhooks' and includes an 'Add webhook' button. Below the title, a description states: 'Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).' A table lists the configured webhooks. One webhook is shown with the URL 'http://ronjenkins.com/ (push)', which is circled in red and has a green checkmark icon to its left. To the right of the URL are 'Edit' and 'Delete' buttons.

In github, either you create file directly or upload Jenkinsfile from local. This is a hello world Jenkinsfile.

The first screenshot shows the repository 'ronslim/jenkins' on the GitHub website. The 'Code' tab is selected. Below the repository name, there are buttons for 'main' branch, '1 branch', and '0 tags'. A list of recent commits is shown, including 'ronslim Update README.md'. Below the commits, a list of files is displayed: 'Jenkinsfile' (circled in red) and 'README.md'. The second screenshot shows the 'Jenkinsfile' file selected in the repository. The file content is displayed in a code editor, showing a Jenkins pipeline configuration. The code is as follows:

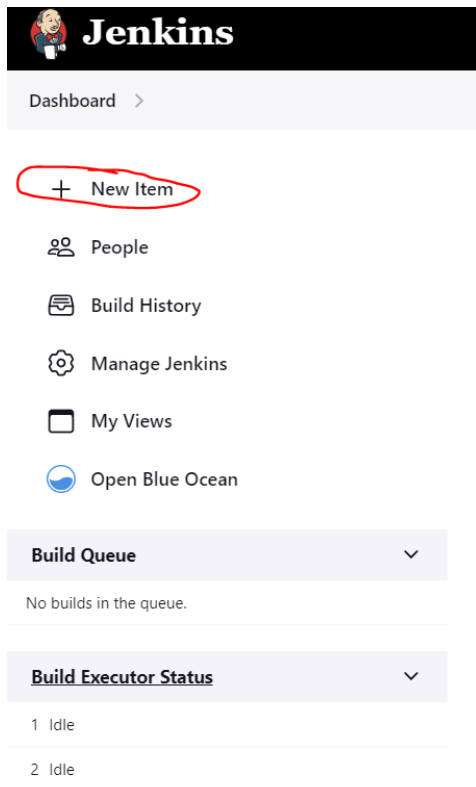
```
1 pipeline {
2   agent any
3   stages {
4     stage('Stage 1') {
5       steps {
6         echo 'Hello world!'
7       }
8     }
9   }
10 }
```

Jenkinsfile YAML code:

```
pipeline {
  agent any
  stages {
    stage('Stage 1') {
      steps {
        echo 'Hello world!'
      }
    }
  }
}
```

For this case, the Github directory of Jenkinsfile is *ronslim/Jenkins/*


Then we are ready for pipeline with Github integration as we already have working webhook and Jenkinsfile to be used in pipeline run. Click + New Item





Then specify an item name and choose Pipeline and then click OK button


**Enter an item name**

ronpipelinegithubintegration  
» Required field

 **Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.


 **Maven project**  
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

 **Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

 **Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific  
Folder

OK

Then in General window, check **Github Project** and under Build Triggers, check **GitHub hook trigger for GITScm polling**. Under **Github Project**, specify Jenkins project folder in Github.

**General** Enabled 

Description

[Plain text] [Preview](#)

☐ Discard old builds ?

☐ Do not allow concurrent builds

☐ Do not allow the pipeline to resume if the controller restarts

☒ **GitHub project**

Project url ?

https://github.com/ronslim/jenkins

## Build Triggers

- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ Build whenever a **SNAPSHOT** dependency is built ?
- ☒ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?
- ☐ Quiet period ?
- ☐ Trigger builds remotely (e.g., from scripts) ?

For the Pipeline section, choose **Pipeline script from SCM** Definition. For the SCM, choose Git and then specify Repository URL which is the same as Project URL. Then use \*/main for Branch specifier just in case \*/master does not work.

### Pipeline

#### Definition

Pipeline script from SCM

#### SCM ?

Git

#### Repositories ?

##### Repository URL ?

https://github.com/ronslim/jenkins

##### Credentials ?

- none -

Add

Then click Save button

(Auto) ▼

Additional Behaviours

Add ▼

Script Path ?

Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

**Save** Apply

Then click Build Now to start with the build

Dashboard > ronpipelinegithubintegration >

Status

</> Changes

**> Build Now**

Configure

Delete Pipeline

Full Stage View

GitHub

Open Blue Ocean

Rename

Pipeline Syntax

GitHub Hook Log

### Pipeline ronpipelinegithubintegration

[Add description](#)

Disable Project

#### Stage View

No data available. This Pipeline has not yet run.

#### Permalinks

Then monitor until success or failure

Status

</> Changes

▶ Build Now

⚙️ Configure

🗑️ Delete Pipeline

🔍 Full Stage View

🌐 GitHub

🌊 Open Blue Ocean

✎ Rename

🔗 Pipeline Syntax

📄 GitHub Hook Log

## Pipeline ronpipelinegithubintegration

### Stage View

|   | Declarative: Checkout SCM | Stage 1 |
|---|---------------------------|---------|
| Average stage times:<br>(Average full run time: ~14s) | 4s                        | 813ms   |
| #1<br>Jun 13 07:55<br>No Changes                      | 4s                        | 813ms   |

### Permalinks

Build History trend ▾

Filter builds... /

✓ #1 Jun 13, 2023, 7:54 AM ↗

Click either #1 or the date beside #1 to check the first build run

Then click Pipeline Steps to check the steps performed in Build #1

Dashboard > ronpipelinegithubintegration > #1

Status

</> Changes

📄 Console Output

✎ Edit Build Information

🗑️ Delete build '#1'

📊 Git Build Data

🌊 Open Blue Ocean

🔄 Restart from Stage

↺ Replay

**📋 Pipeline Steps**

📁 Workspaces

✓ Build #1 (Jun 13, 2023, 7:54:59 AM)

Started by user Administrator

git

Revision: e80b0b43b135bd8efdc16fb9d5bc4737f67409f3  
Repository: <https://github.com/ronslim/jenkins>

- refs/remotes/origin/main



After checking the pipeline steps performed, click Console Output to check the output logs

The screenshot shows the Jenkins web interface. On the left, a sidebar contains navigation links: Status, </> Changes, Console Output (highlighted with a red circle), Edit Build Information, Delete build '#1', Git Build Data, Open Blue Ocean, Restart from Stage, Replay, Pipeline Steps, and Workspaces. The main area displays a table of pipeline steps:

| Step   | Arguments                       | Status |
|--|---------------------------------|--------|
| Start of Pipeline - (11 sec in block)                      |                                 | ✓      |
| node - (8.3 sec in block)                                  |                                 | ✓      |
| node block - (7.3 sec in block)                            |                                 | ✓      |
| stage - (4.9 sec in block)                                 | Declarative: Checkout SCM       | ✓      |
| stage block (Declarative: Checkout SCM) - (4 sec in block) |                                 | ✓      |
| checkout - (3.5 sec in self)                               |                                 | ✓      |
| withEnv - (1.8 sec in block)                               | GIT_BRANCH, GIT_COMMIT, GIT_URL | ✓      |
| withEnv block - (1.4 sec in block)                         |                                 | ✓      |
| stage - (0.84 sec in block)                                | Stage 1                         | ✓      |
| stage block (Stage 1) - (0.29 sec in block)                |                                 | ✓      |
| echo - (0.13 sec in self)                                  | Hello world!                    | ✓      |

Below the table, the 'Console Output' tab is selected, showing a green checkmark icon and the title 'Console Output'. The output text is as follows:

```
Started by user Administrator
Obtained Jenkinsfile from git https://github.com/ronslim/jenkins
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in D:\Jenkins\jenkins\workspace\ronpipelinegithubintegration
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/ronslim/jenkins
> git.exe init D:\Jenkins\jenkins\workspace\ronpipelinegithubintegration # timeout=10
Fetching upstream changes from https://github.com/ronslim/jenkins
> git.exe --version # timeout=10
> git --version # 'git version 2.31.1.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/ronslim/jenkins +refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe config remote.origin.url https://github.com/ronslim/jenkins # timeout=10
> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
```

Console Output (Full):

Started by user [Administrator](#)  
Obtained Jenkinsfile from git <https://github.com/ronslim/jenkins>  
[Pipeline] Start of Pipeline  
[Pipeline] node  
Running on [Jenkins](#) in D:\Jenkins\jenkins\workspace\ronpipelinegithubintegration  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Declarative: Checkout SCM)  
[Pipeline] checkout  
Selected Git installation does not exist. Using Default  
The recommended git tool is: NONE  
No credentials specified  
Cloning the remote Git repository  
Cloning repository <https://github.com/ronslim/jenkins>

```
> git.exe init D:\Jenkins\jenkins\workspace\ronpipelinegithubintegration # timeout=10
Fetching upstream changes from https://github.com/ronslim/jenkins
> git.exe --version # timeout=10
> git --version # 'git version 2.31.1.windows.1'
> git.exe fetch --tags --force --progress -- https://github.com/ronslim/jenkins
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git.exe config remote.origin.url https://github.com/ronslim/jenkins # timeout=10
> git.exe config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision e80b0b43b135bd8efdc16fb9d5bc4737f67409f3 (refs/remotes/origin/main)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f e80b0b43b135bd8efdc16fb9d5bc4737f67409f3 # timeout=10
Commit message: "Update README.md"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Stage 1)
[Pipeline] echo
Hello world!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```