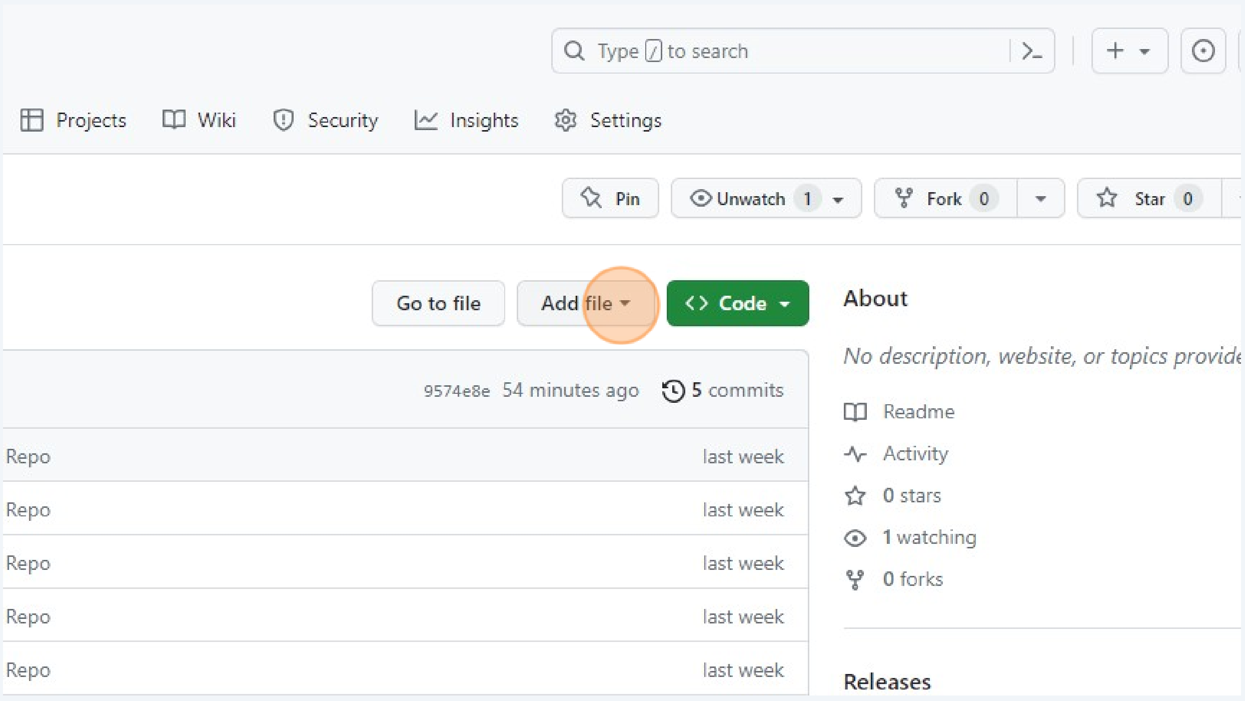
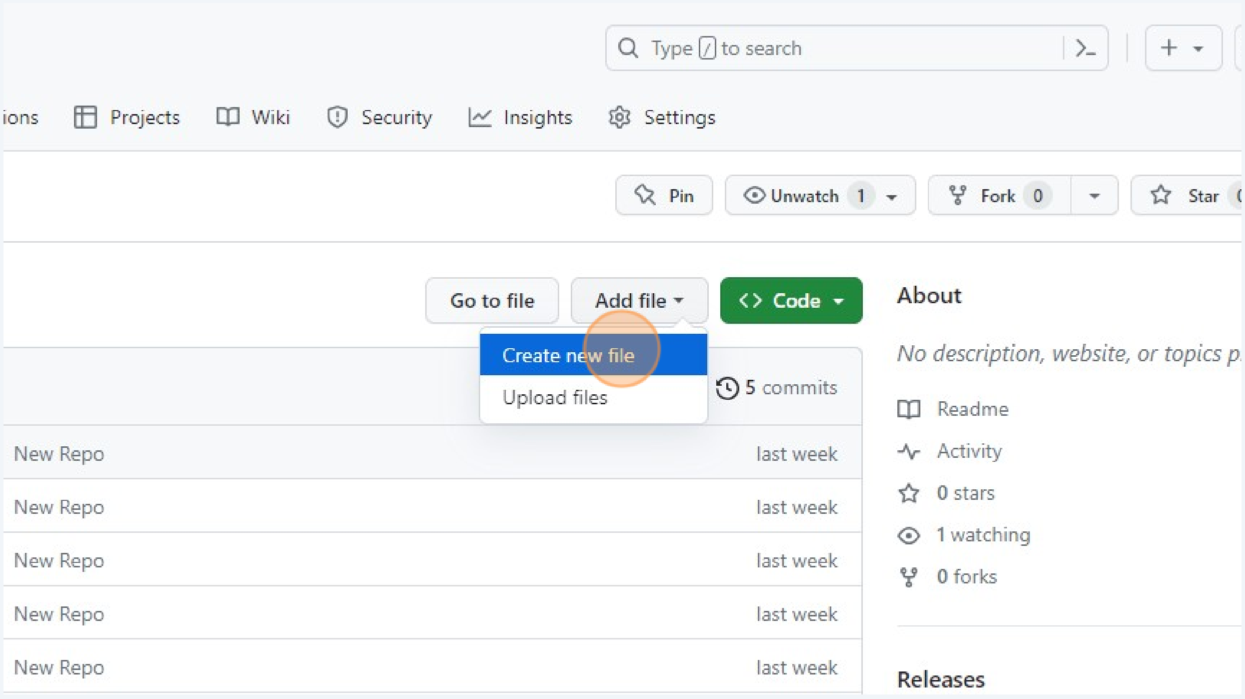
Setting Up Jenkins Pipeline for React Sample App Deployment

1 Now we need to start our pipeline, for that lets go to github to our repository.

2 On the repo we need to add our Jenkins Pipeline. You can either push a file or directly add the content here by Clicking "Add file"

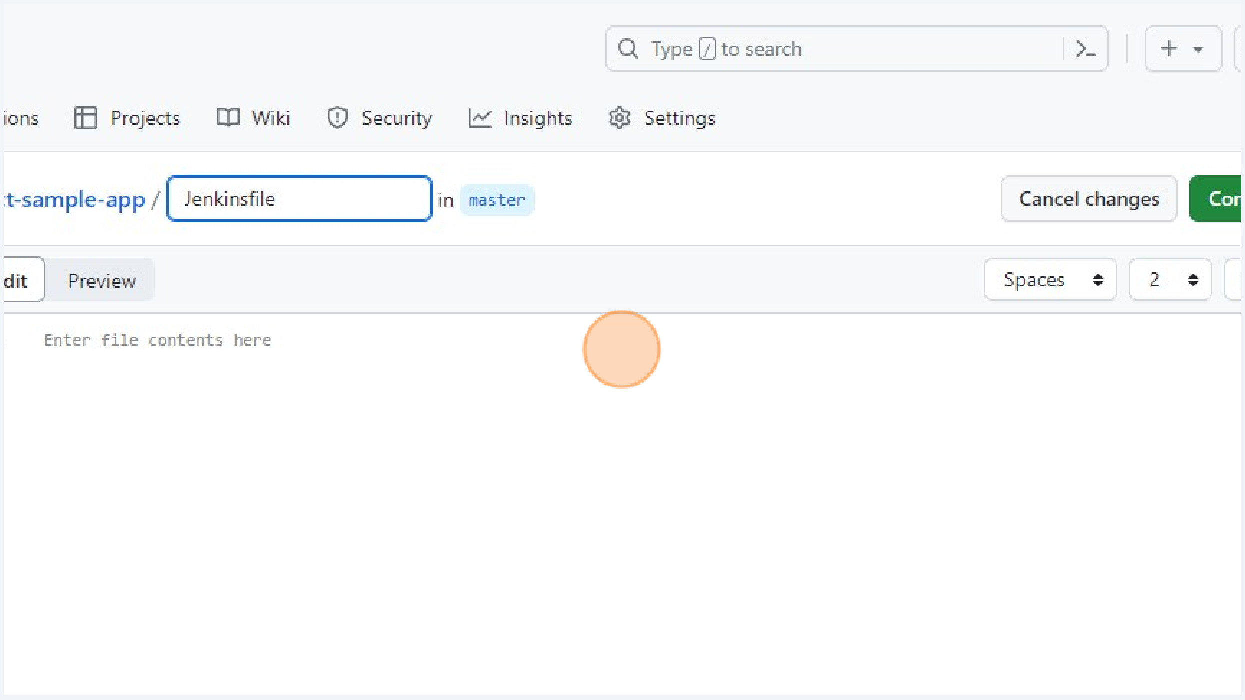


3 Click "Create new file"



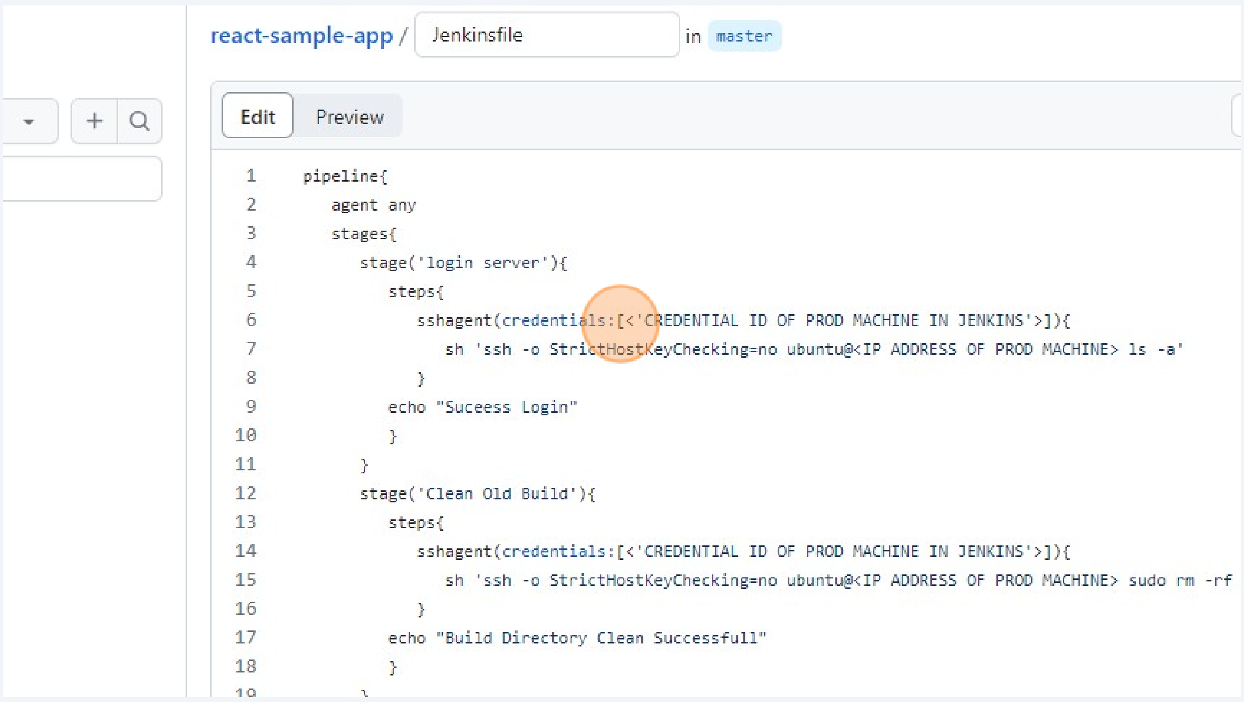
New Repo	last week
New Repo	last week
New Repo	last week
New Repo	last week
New Repo	last week

4 Make sure that the file name is "Jenkinsfile" or name that you should remember while setting in Jenkins Pipeline configuration.



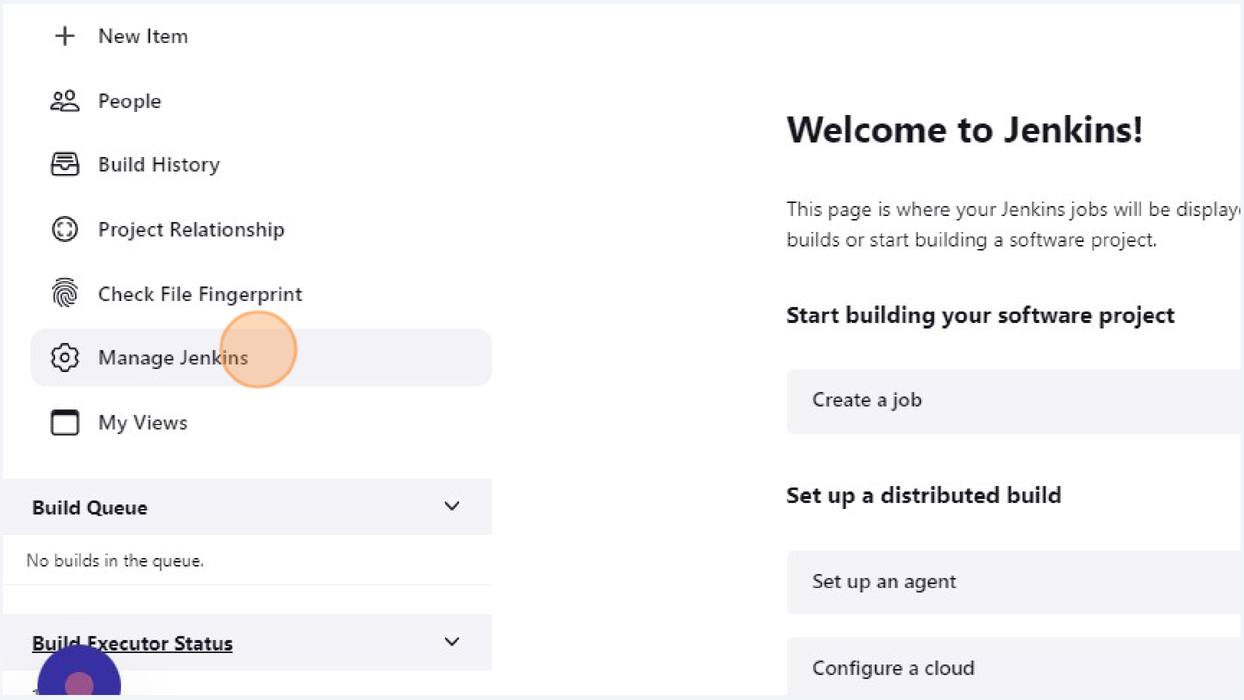
5 The Pipeline Syntax is available inside the Jenkins folder, either modify that and upload or copy paste here and follow the further steps.

6 Here on the file we need to add the credential ID and IP address of our production machine.



```
pipeline{
   agent any
   stages{
      stage('login server'){
         steps{
            sshagent(credentials:[<'CREDENTIAL ID OF PROD MACHINE IN JENKINS'>]){
               sh 'ssh -o StrictHostKeyChecking=no ubuntu@<IP ADDRESS OF PROD MACHINE> ls -a'
            }
         echo "Suceess Login"
         }
      }
      stage('Clean Old Build'){
         steps{
            sshagent(credentials:[<'CREDENTIAL ID OF PROD MACHINE IN JENKINS'>]){
               sh 'ssh -o StrictHostKeyChecking=no ubuntu@<IP ADDRESS OF PROD MACHINE> sudo rm -rf
            }
         echo "Build Directory Clean Successfull"
         }
```

7 To know what is your Credential ID go to jenkins and click Manage Jenkins



Welcome to Jenkins!

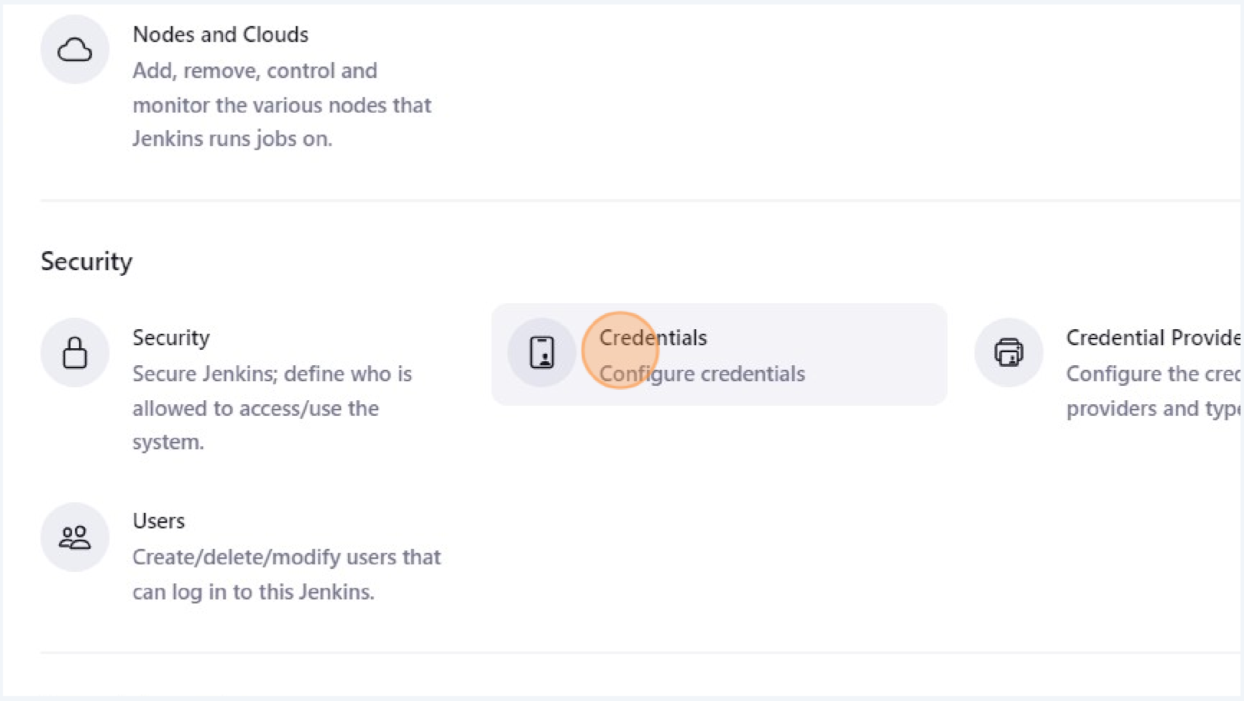
This page is where your Jenkins jobs will be display
builds or start building a software project.

Start building your software project

Set up a distributed build

No builds in the queue.

8 Click "Credentials"



Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

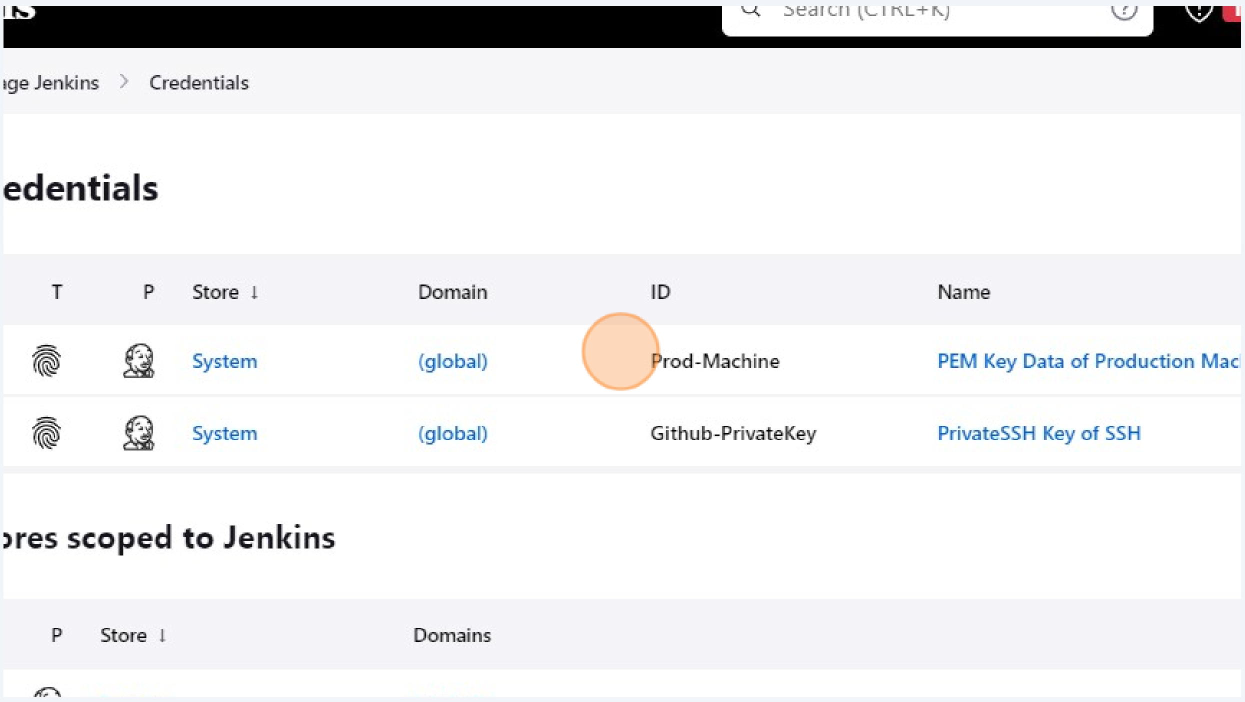
Secure Jenkins; define who is allowed to access/use the system.

Configure credentials

Configure the cred
providers and typ

Create/delete/modify users that can log in to this Jenkins.

9 I stored the PEM file key of my production machine under the ID "Prod-Machine". So copy that as this is our ID.


ıge Jenkins > Credentials

edentials

T	P	Store ↓	Domain	ID	Name
		System	(global)	Prod-Machine	PEM Key Data of Production Mac
		System	(global)	Github-PrivateKey	PrivateSSH Key of SSH

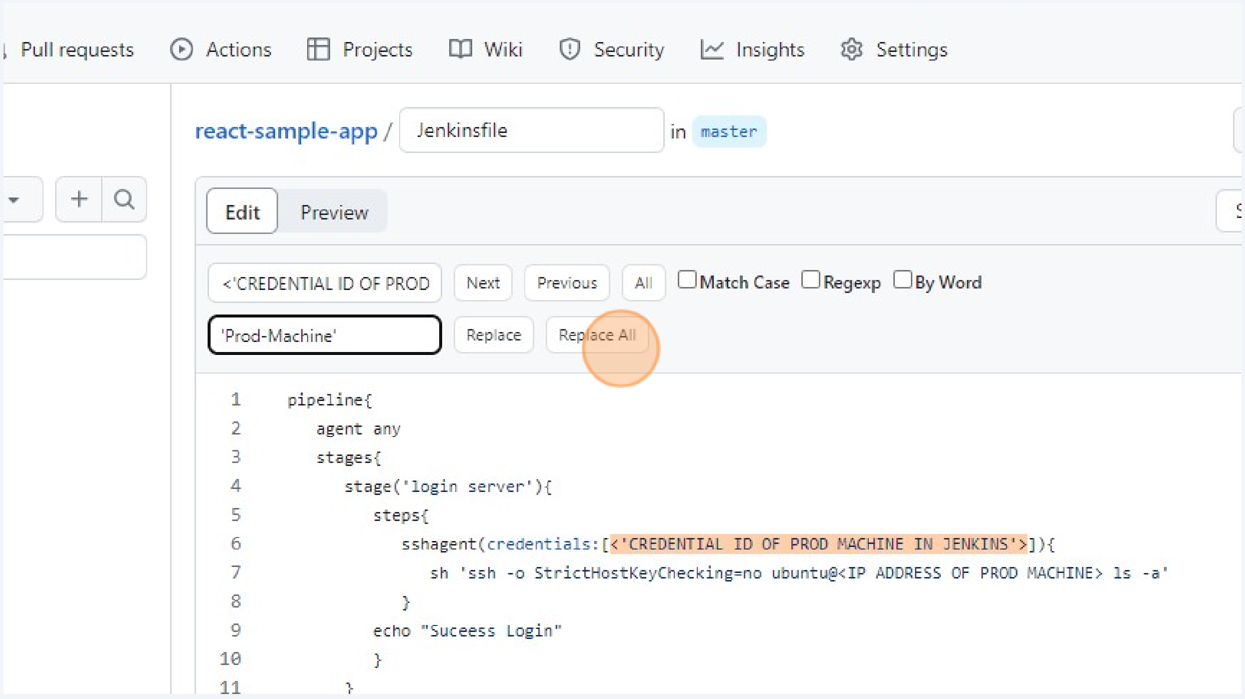
ɔres scoped to Jenkins

P	Store ↓	Domains

10 Go back to editing the pipeline tab. A Simple hack here is to click Press **CTRL** + **F**

11 Now add the contents for find and replace field. In the find field it will be <'CREDENTIAL ID OF PROD MACHINE IN JENKINS'> and on the replace field it must be 'Prod-Machine' . Make sure to have single quotes for Prod-Machine.

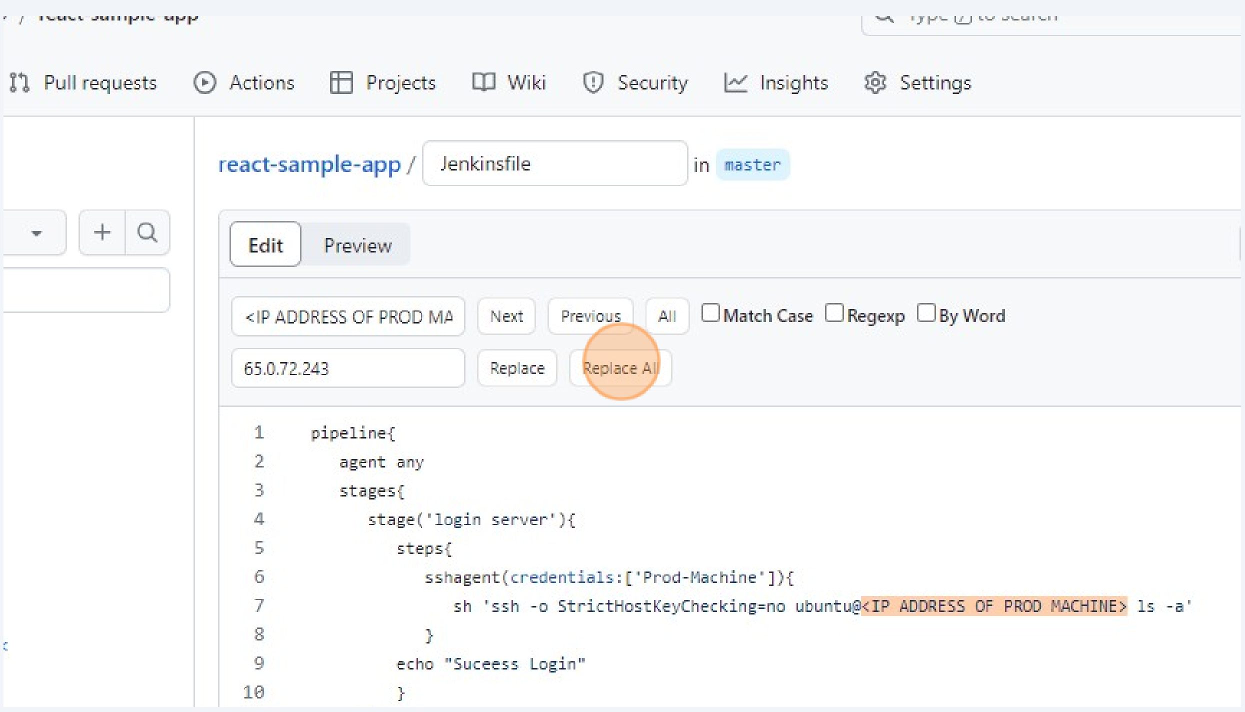
12 After that, Click "Replace All"



```
pipeline{
   agent any
   stages{
      stage('login server'){
         steps{
            sshagent(credentials:[<'CREDENTIAL ID OF PROD MACHINE IN JENKINS'>]){
               sh 'ssh -o StrictHostKeyChecking=no ubuntu@<IP ADDRESS OF PROD MACHINE> ls -a'
            }
         echo "Suceess Login"
         }
      }
```

13 Now we need to do the same step again for setting the IP address. For that in the find field it will be <IP ADDRESS OF PROD MACHINE> and replace field it will be the IP address of your machine.

14 Click "Replace All"

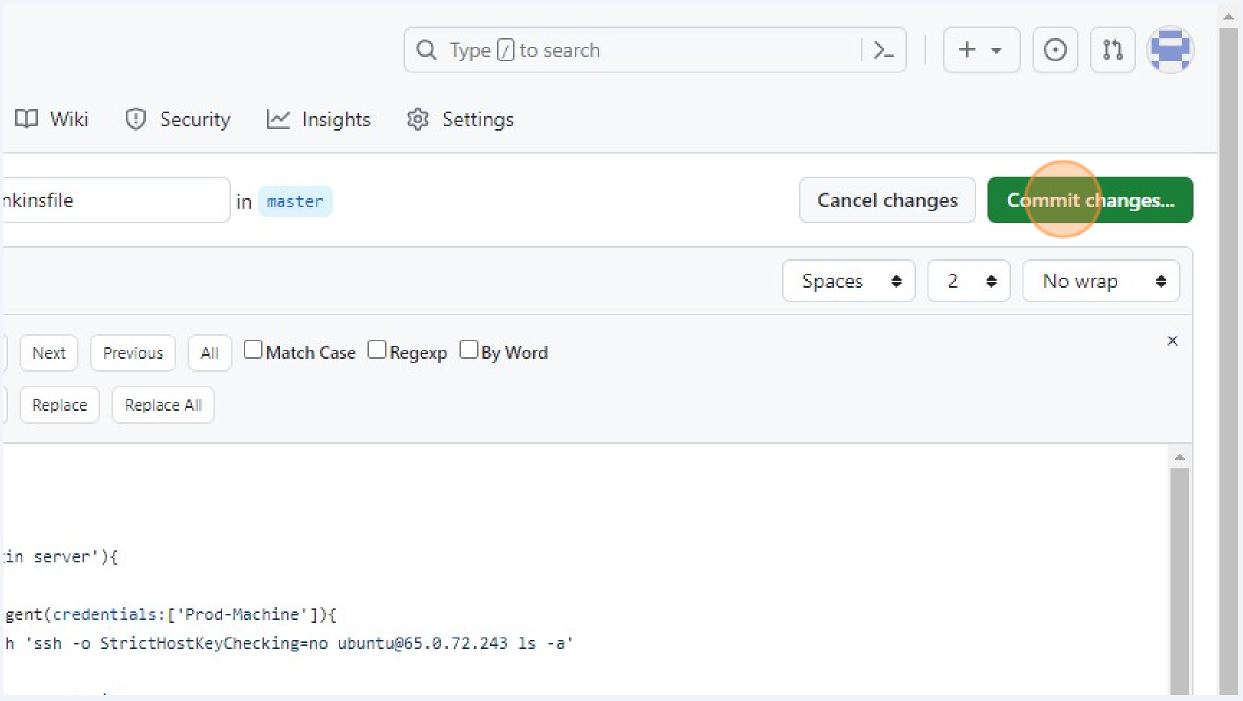


<IP ADDRESS OF PROD MA

65.0.72.243

```
pipeline{
   agent any
   stages{
      stage('login server'){
         steps{
            sshagent(credentials:['Prod-Machine']){
               sh 'ssh -o StrictHostKeyChecking=no ubuntu@<IP ADDRESS OF PROD MACHINE> ls -a'
            }
         echo "Suceess Login"
         }
```

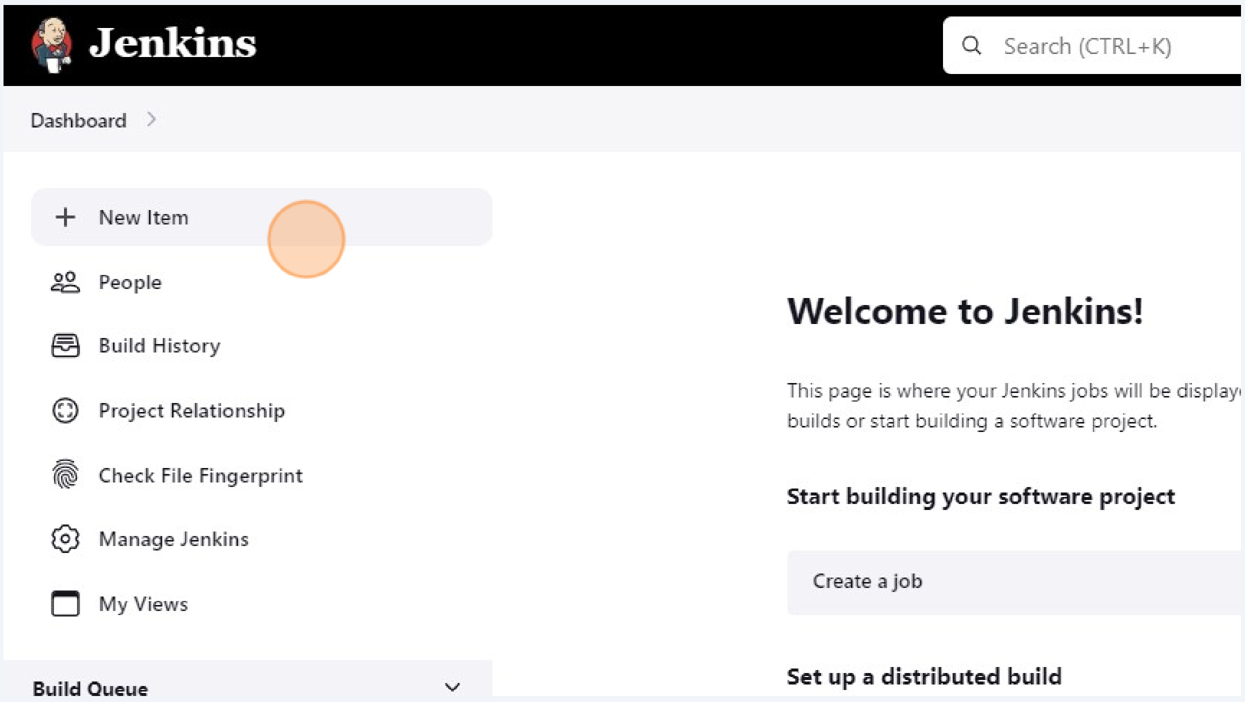
15 Click "Commit changes..."



```
in server'){

gent(credentials:['Prod-Machine']){
h 'ssh -o StrictHostKeyChecking=no ubuntu@65.0.72.243 ls -a'
```

16 Now go back to Jenkins and Click "New Item"



Welcome to Jenkins!

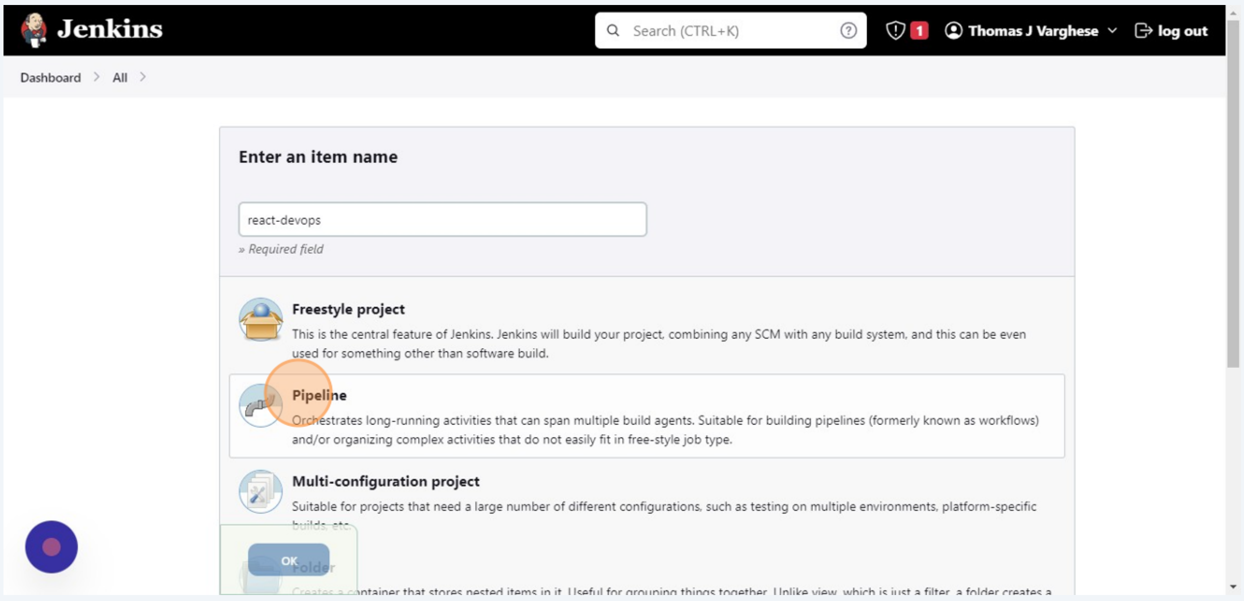
This page is where your Jenkins jobs will be display builds or start building a software project.

Start building your software project

Create a job

Set up a distributed build

17 Enter a name and Click "Pipeline" and then click "OK"



Enter an item name

react-devops

» Required field

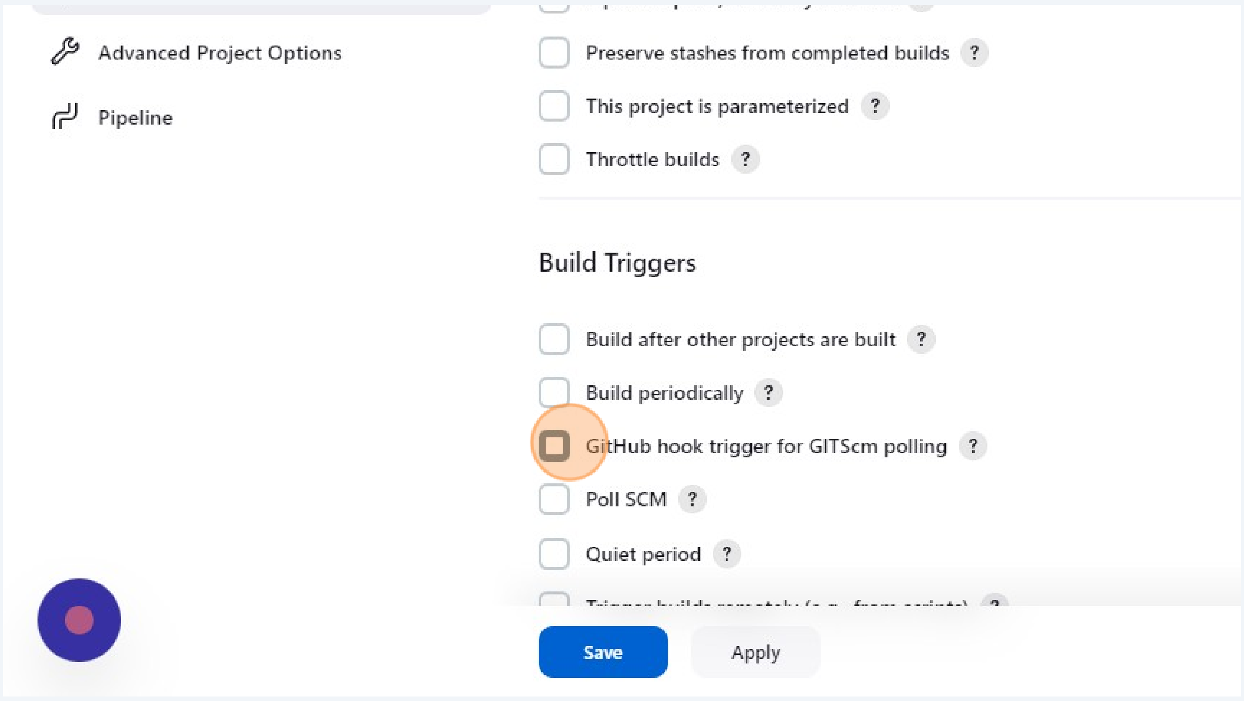
Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

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 builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a

18 Scroll a bit down and check "GitHub hook trigger for GITScm polling" under Build Triggers



Preserve stashes from completed builds ?

This project is parameterized ?

Throttle builds ?

Build Triggers

Build after other projects are built ?

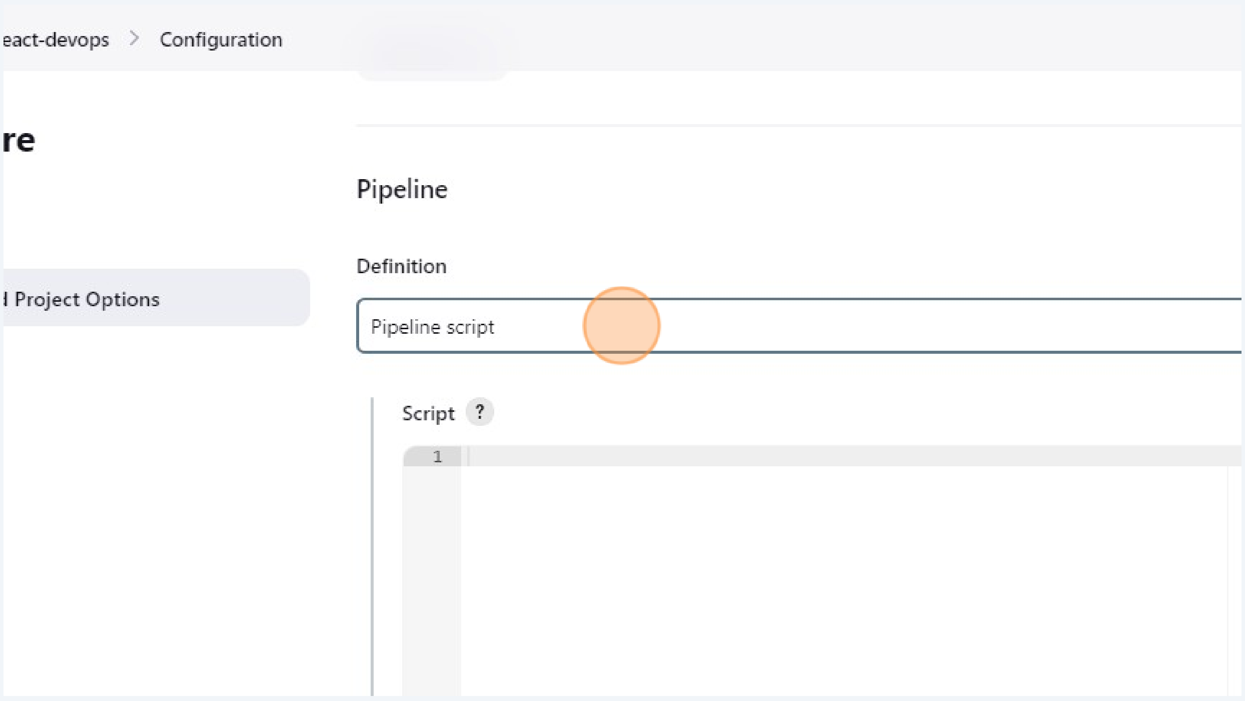
Build periodically ?

GitHub hook trigger for GITScm polling ?

Poll SCM ?

Quiet period ?

19 Go down and Click this dropdown.



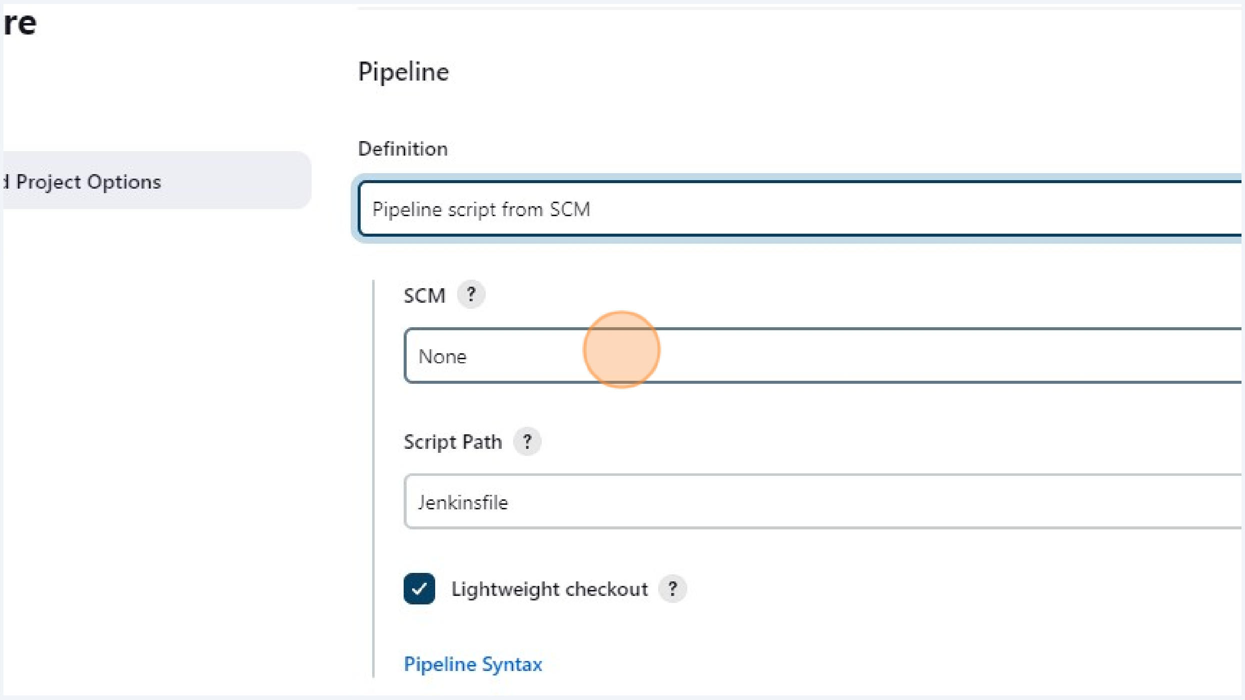
Pipeline

Definition

Pipeline script

Script ?

20 Select the second one as below and in SCM select Git


re

Project Options

Pipeline

Definition

Pipeline script from SCM

SCM ?

None

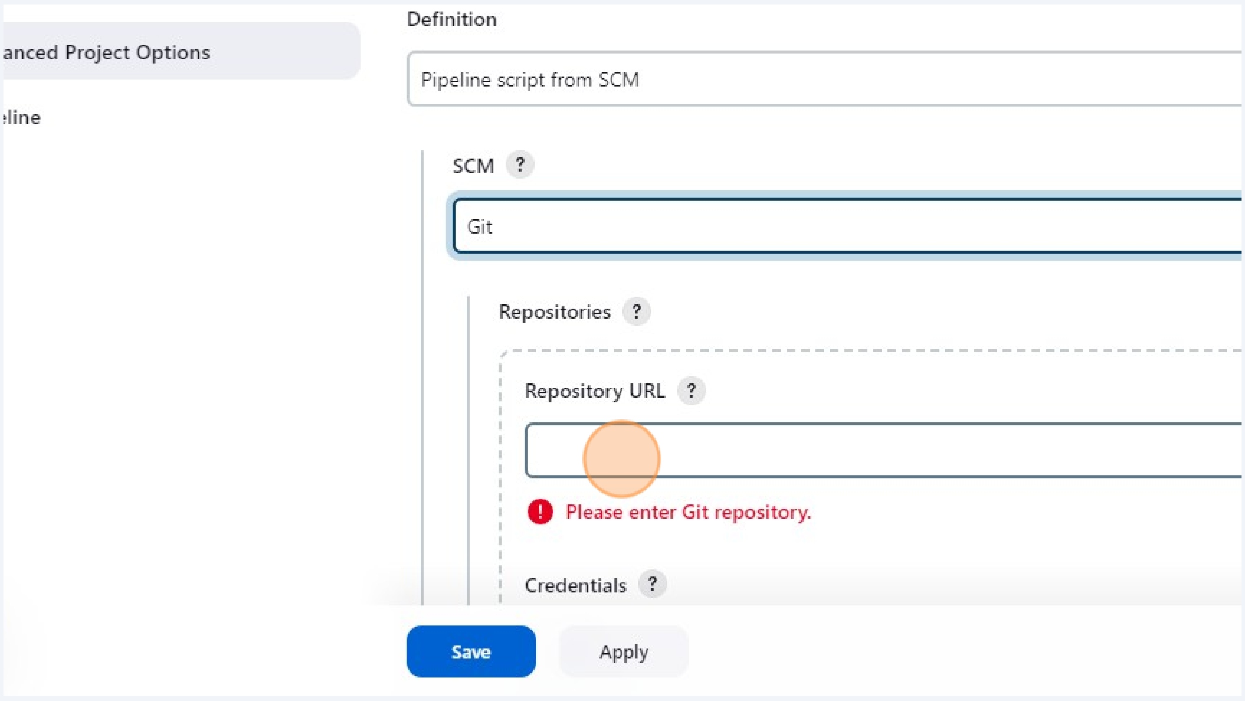
Script Path ?

Jenkinsfile

✓ Lightweight checkout ?

Pipeline Syntax

21 On the repository URL, give github URL of the repo


anced Project Options

line

Definition

Pipeline script from SCM

SCM ?

Git

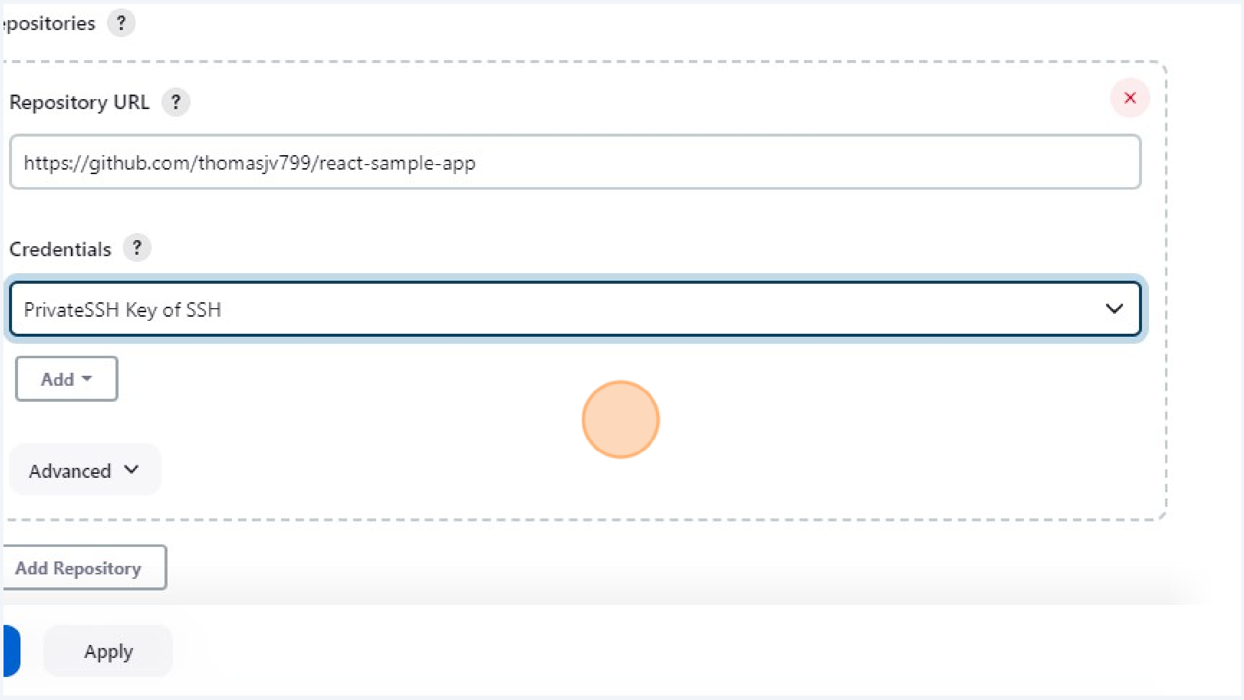
Repositories ?

Repository URL ?

Please enter Git repository.

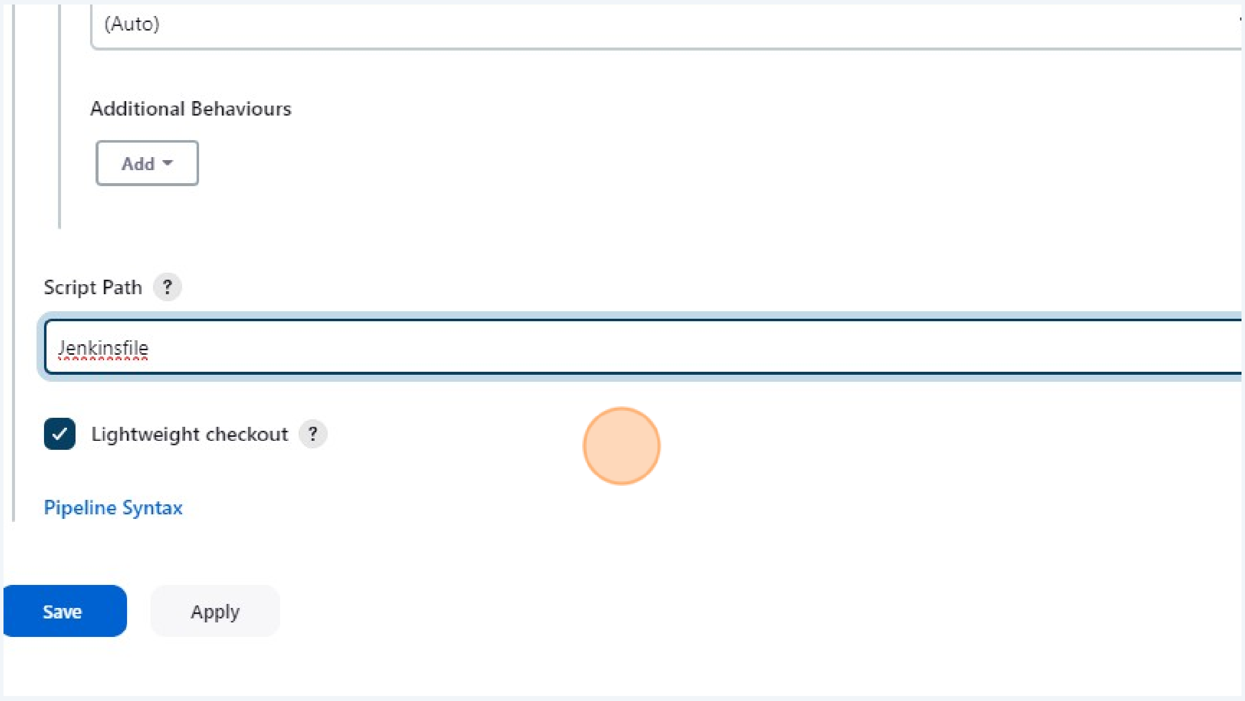
Credentials ?

22 If it is private, we need to select the credential that we saved earlier with the private SSH key of our Production Machine.


Repository URL: https://github.com/thomasjv799/react-sample-app

Credentials: PrivateSSH Key of SSH

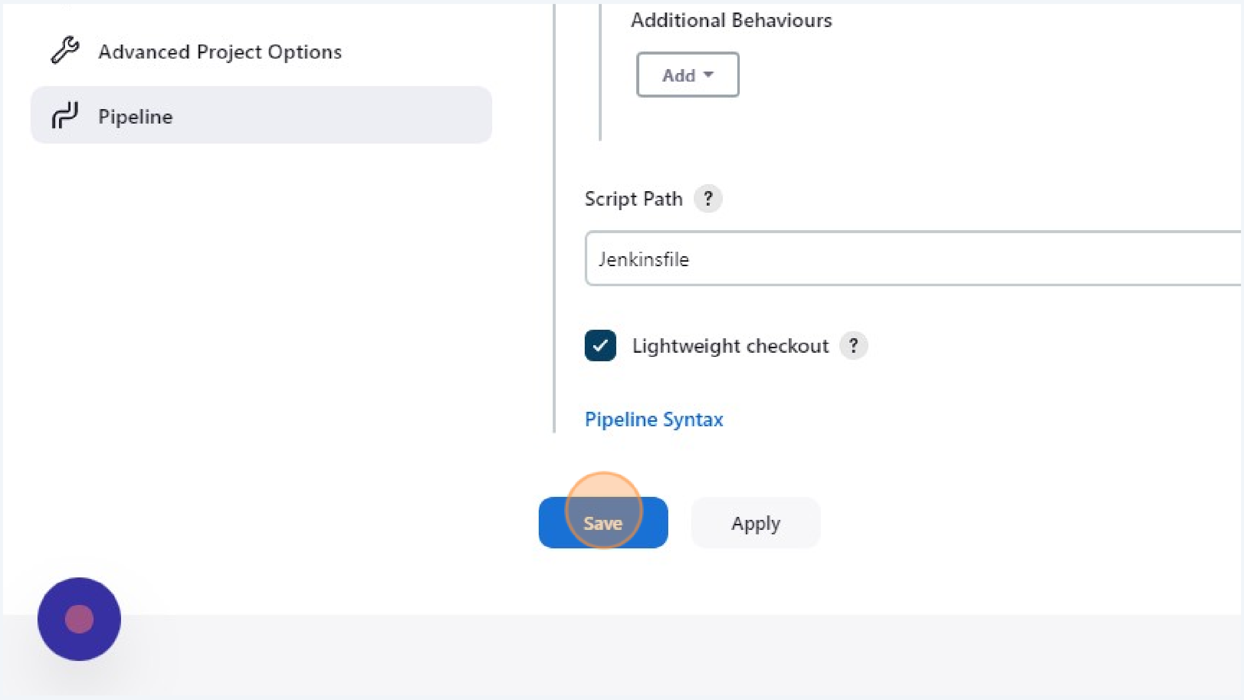
23 You can check the rest of the fields. By default everything is set.


(Auto)

Script Path: Jenkinsfile

✓ Lightweight checkout

24 Click "Save"


Advanced Project Options

Pipeline

Additional Behaviours

Add

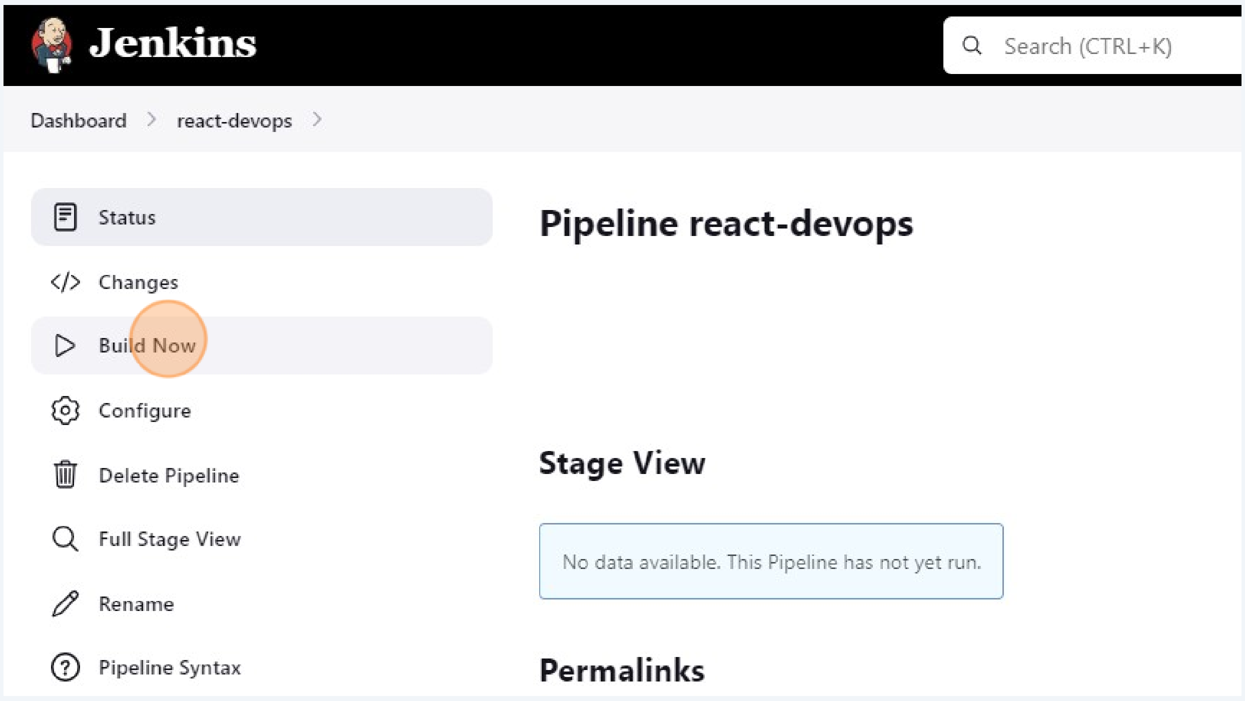
Script Path ?

Jenkinsfile

✓ Lightweight checkout ?

Pipeline Syntax

25 Click "Build Now" and wait for a few seconds.



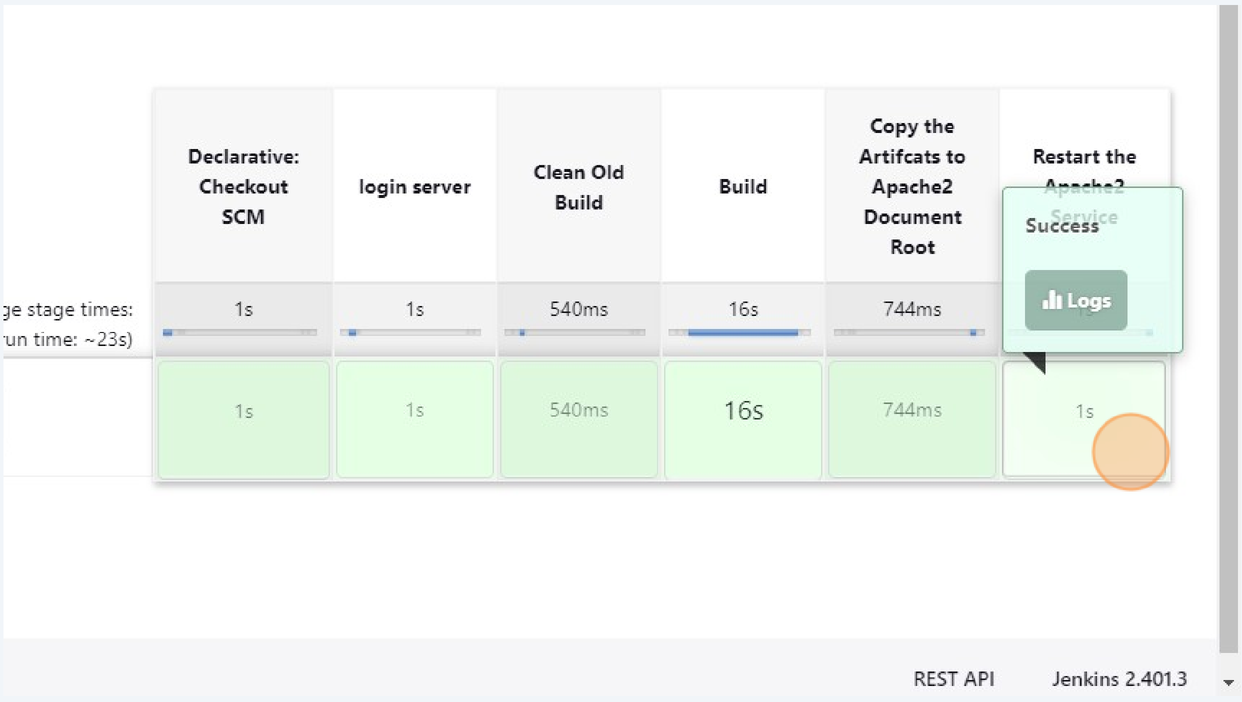
Pipeline react-devops

Stage View

No data available. This Pipeline has not yet run.

Permalinks

26 If everything is set correctly then Jenkins will run the entire pipeline successfully. Next time if you commit a change to the master branch of your repo, then will pipeline will automatically trigger the entire pipeline and deploy in the production machine.



	Declarative: Checkout SCM	login server	Clean Old Build	Build	Copy the Artifcats to Apache2 Document Root	Restart the Apache2 Service
ge stage times: un time: ~23s)	1s	1s	540ms	16s	744ms	1s
	1s	1s	540ms	16s	744ms	1s