

# **Dvs Technologies Aws & Devops**

Compiled and Scrutinized by  
**Mr. Shaan Shaik**  
(Senior DevOps Lead)

## **Words To The Students**

Though we have taken utmost efforts to present you this book error free, but still it may contain some errors or mistakes. Students are encouraged to bring, if there are any mistakes or errors in this document to our notice. So that it may be rectified in the next edition of this document.

**“Suppressing your doubts is Hindering your growth”.**

We urge you to work hard and make use of the facilities we are providing to you, because there is no substitute for hard work. We wish you all the best for your future.

**“The grass isn’t greener on the other side; the grass is greener where you water it.”**

You and your suggestions are valuable to us; Help us to serve you better. In case of any suggestions, grievance, or complaints, please feel free to write us your suggestions, grievance and feedback on the following

**Dvs.training@gmail.com**

# **Git and Git Hub Learning::**

We have many version control software's in market. They are as follows

1. CVS
2. PVCS
3. Subversion(svn)
4. Perforce
5. Microsoft Visual Sourcesafe
6. Mercurial
7. Teamsite
8. vault
9. Bitkeeper - Used to manage the linux kernel before
10. Git - Created by Linus author of linux

**Different Phases as part of Git :**

**Initialization:**

creating the empty repository for use

**Clone:**

Making a local full copy on your workstation

**Checking out:**

Locking a copy of one or more files for exclusive use

**Branching:**

Allowing a set of files to be developed concurrently and at different speeds for different reasons

**Merging:**

Taking different branches or sets of changes and integrating into one set or branch

**Resolving:**

Taking conflicting changes from multiple people on the same file and manually addressing

**Commit:**

Taking changes from the local system and comiting them to the branch

**Push/Pull:**

Taking changes locally or remotely and merging into one or more branches

## Installation:

### yum install git -y

```
[root@ip-172-31-9-43 ~]# yum install git -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
192 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
--> Package git.x86_64 0:2.23.3-1.amzn2.0.1 will be installed
--> Processing Dependency: perl-Git = 2.23.3-1.amzn2.0.1 for package: git-2.23.3-1.amzn2.0.1.x86_64
--> Processing Dependency: git-core-doc = 2.23.3-1.amzn2.0.1 for package: git-2.23.3-1.amzn2.0.1.x86_64
--> Processing Dependency: git-core = 2.23.3-1.amzn2.0.1 for package: git-2.23.3-1.amzn2.0.1.x86_64
--> Processing Dependency: emacsfilesystem >= 25.3 for package: git-2.23.3-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(Term::ReadKey) for package: git-2.23.3-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(Git::T18N) for package: git-2.23.3-1.amzn2.0.1.x86_64
```

### [root@node2 ~]# git --version git version 2.7.5

```
[root@ip-172-31-9-43 mytest]# git --version
git version 2.23.3
[root@ip-172-31-9-43 mytest]#
```

## Configuring User and email:

### ~/.gitconfig

```
[root@ip-172-31-9-43 mytest]# git config --global user.name "gituserid"
[root@ip-172-31-9-43 mytest]# git config --global user.email "shahan.aix@gmail.com"
[root@ip-172-31-9-43 mytest]# git config --list
user.name=gituserid
user.email=shahan.aix@gmail.com
[root@ip-172-31-9-43 mytest]# cat ~/.gitconfig
[user]
    name = gituserid
    email = shahan.aix@gmail.com
[root@ip-172-31-9-43 mytest]#
```

## Basic Configuration:

### /etc/gitconfig

**We have three things to be considered here**

**--local :-> Deals with the local repository**

**--global:-> Deals with the global repository**

**--system :-> Deals with the system repository**

**configure the editor for our git**

```
[root@ip-172-31-9-43 mytest]# git config --system system.editor vim
[root@ip-172-31-9-43 mytest]# git config --system system.name "mysystemname"
[root@ip-172-31-9-43 mytest]# cat /etc/gitconfig
[system]
    editor = vim
    name = mysystemname
[root@ip-172-31-9-43 mytest]# git config --list
system.editor=vim
system.name=mysystemname
user.name=gituserid
user.email=shahan.aix@gmail.com
[root@ip-172-31-9-43 mytest]#
```

**Git Basics :**

**Empty Repositories:**

**Initializing empty folder as git repos with "git init ." command as specified below. Before initialization we don't have anything called ".git" but post initialization we can see that ".git" folder got created.**

```

[root@ip-172-31-9-43 tmp]# mkdir myrepo
[root@ip-172-31-9-43 tmp]# cd myrepo/
[root@ip-172-31-9-43 myrepo]# ls -al
total 0
drwxr-xr-x 2 root root 6 Jul 27 18:05 .
drwxrwxrwt 12 root root 328 Jul 27 18:05 ..
[root@ip-172-31-9-43 myrepo]# git init .
Initialized empty Git repository in /tmp/myrepo/.git/
[root@ip-172-31-9-43 myrepo]# ls -al
total 0
drwxr-xr-x 3 root root 18 Jul 27 18:05 .
drwxrwxrwt 12 root root 328 Jul 27 18:05 ..
drwxr-xr-x 7 root root 119 Jul 27 18:05 .git
[root@ip-172-31-9-43 myrepo]# ls -la .git/
total 12
drwxr-xr-x 7 root root 119 Jul 27 18:05 .
drwxr-xr-x 3 root root 18 Jul 27 18:05 ..
drwxr-xr-x 2 root root 6 Jul 27 18:05 branches
-rw-r--r-- 1 root root 92 Jul 27 18:05 config
-rw-r--r-- 1 root root 73 Jul 27 18:05 description
-rw-r--r-- 1 root root 23 Jul 27 18:05 HEAD
drwxr-xr-x 2 root root 301 Jul 27 18:05 hooks
drwxr-xr-x 2 root root 21 Jul 27 18:05 info
drwxr-xr-x 4 root root 30 Jul 27 18:05 objects
drwxr-xr-x 4 root root 31 Jul 27 18:05 refs
[root@ip-172-31-9-43 myrepo]#

```

## Performing Operations:

We are going to perform below operations as part of our git

```

total 0
[root@ip-172-31-9-43 myrepo]# touch test(1..4).txt
[root@ip-172-31-9-43 myrepo]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:09 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:09 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:09 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:09 test4.txt

```

```

[root@ip-172-31-9-43 myrepo]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        test1.txt
        test2.txt
        test3.txt
        test4.txt

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 myrepo]# git add .
[root@ip-172-31-9-43 myrepo]# git commit -m "adding test files"
[master (root-commit) d557bf5] adding test files
 4 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 test1.txt
 create mode 100644 test2.txt
 create mode 100644 test3.txt
 create mode 100644 test4.txt
[root@ip-172-31-9-43 myrepo]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-9-43 myrepo]# git log
commit d557bf55b2d4d44aaa4334fd3657fe17210c3b49 (HEAD -> master)
Author: gituserid <shahan.aix@gmail.com>
Date:   Mon Jul 27 18:09:55 2020 +0000

    adding test files

```

## Git Ignore:

### Before adding the ".gitignore" file

```

[root@ip-172-31-9-43 myrepo]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-9-43 myrepo]# touch myscript.sh
[root@ip-172-31-9-43 myrepo]# git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        myscript.sh

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 myrepo]#

```

**Post adding ".gitignore" file lets see if our git add/recognizes the myscript.sh file or not.**

```
nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 myrepo]# echo "*.sh" > .gitignore
[root@ip-172-31-9-43 myrepo]# git status
On branch master
Untracked files:
  (use "git add <file>" to include in what will be committed)
    .gitignore

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 myrepo]# cat .gitignore
*.sh
[root@ip-172-31-9-43 myrepo]#
```

**From the above output we can say that soon after we add our "\*.sh" to our ".gitignore" file it's simply ignoring my files ending with "\*.sh" files in our case its "myscript.sh"**

### Normal copy operation

```
[root@ip-172-31-9-43 tmp]# mkdir cpsrc cpdest
[root@ip-172-31-9-43 tmp]# touch cpsrc/test{1..4}.txt
[root@ip-172-31-9-43 tmp]# ls -l cpsrc/
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:21 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:21 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:21 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:21 test4.txt
[root@ip-172-31-9-43 tmp]# ls -l cpdest/
total 0
```

### Post copy operation

```
[root@ip-172-31-9-43 tmp]# cp -rf cpsrc/* cpdest/
[root@ip-172-31-9-43 tmp]# ls -l cpdest/
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:24 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:24 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:24 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:24 test4.txt
[root@ip-172-31-9-43 tmp]#
```

**Main drawback is if dest "localdest" is having update it can't sync directly with my source "localsrc". I have to do it manually by executing commands.**

## Cloning: Local Repositories:

Perform the below operation in the src folder as below

```
[root@ip-172-31-9-43 tmp]# mkdir src
[root@ip-172-31-9-43 tmp]# cd src/
[root@ip-172-31-9-43 src]# git init .
Initialized empty Git repository in /tmp/src/.git/
[root@ip-172-31-9-43 src]# touch test{1..4}.txt
[root@ip-172-31-9-43 src]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:29 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:29 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:29 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:29 test4.txt
[root@ip-172-31-9-43 src]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    test1.txt
    test2.txt
    test3.txt
    test4.txt

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 src]# git add .
[root@ip-172-31-9-43 src]# git commit -m "testing git local clone"
[master (root-commit) 489f120] testing git local clone
 4 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 test1.txt
 create mode 100644 test2.txt
 create mode 100644 test3.txt
 create mode 100644 test4.txt
[root@ip-172-31-9-43 src]#
```

Now let's clone the "src" to our destination folder i.e, "dest" as below

```
[root@ip-172-31-9-43 src]# ls -ld /tmp/dest
ls: cannot access /tmp/dest: No such file or directory
[root@ip-172-31-9-43 src]# git clone /tmp/src /tmp/dest
Cloning into '/tmp/dest'...
done.
[root@ip-172-31-9-43 src]# ls -ld /tmp/dest
drwxr-xr-x 3 root root 86 Jul 27 18:32 /tmp/dest
[root@ip-172-31-9-43 src]#
```



Let's do the modification in the destination folder and try to sync it from src as below.

```
[root@ip-172-31-9-43 dest]# pwd
/tmp/dest
[root@ip-172-31-9-43 dest]# touch mytest{1..4}
[root@ip-172-31-9-43 dest]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:34 mytest1
-rw-r--r-- 1 root root 0 Jul 27 18:34 mytest2
-rw-r--r-- 1 root root 0 Jul 27 18:34 mytest3
-rw-r--r-- 1 root root 0 Jul 27 18:34 mytest4
-rw-r--r-- 1 root root 0 Jul 27 18:32 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:32 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:32 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:32 test4.txt
[root@ip-172-31-9-43 dest]# git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    mytest1
    mytest2
    mytest3
    mytest4

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 dest]# git add .
[root@ip-172-31-9-43 dest]# git commit -m "adding data in dest folder"
[master ef5224d] adding data in dest folder
4 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 mytest1
create mode 100644 mytest2
create mode 100644 mytest3
create mode 100644 mytest4
```

**Sync from src folder as follows:**

```
[root@ip-172-31-9-43 src]# git status
On branch master
nothing to commit, working tree clean
[root@ip-172-31-9-43 src]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:29 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:29 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:29 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:29 test4.txt
[root@ip-172-31-9-43 src]# git pull /tmp/dest/
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 2 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (2/2), done.
From /tmp/dest
 * branch          HEAD          -> FETCH_HEAD
Updating 489f120..ef5224d
Fast-forward
 mytest1 | 0
 mytest2 | 0
 mytest3 | 0
 mytest4 | 0
4 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 mytest1
create mode 100644 mytest2
create mode 100644 mytest3
create mode 100644 mytest4
[root@ip-172-31-9-43 src]# ls -l
```

## Branching, Merging and Tagging:

### Branches:

**Master branch is the default branch which will gets initialized when you create the repository and initialize it.**

```
[root@ip-172-31-9-43 tmp]# mkdir mybranchrepo
[root@ip-172-31-9-43 tmp]# cd mybranchrepo/
[root@ip-172-31-9-43 mybranchrepo]# git init .
Initialized empty Git repository in /tmp/mybranchrepo/.git/
[root@ip-172-31-9-43 mybranchrepo]# touch test.txt
[root@ip-172-31-9-43 mybranchrepo]# git add .
[root@ip-172-31-9-43 mybranchrepo]# git commit -m "working with branches"
[master (root-commit) 8fb80f8] working with branches
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 test.txt
[root@ip-172-31-9-43 mybranchrepo]# git branch
* master
[root@ip-172-31-9-43 mybranchrepo]# git checkout -b branch1
Switched to a new branch 'branch1'
[root@ip-172-31-9-43 mybranchrepo]# git branch
* branch1
  master
```

## Let's check it out what data do we have in branch1 & master branch

```
[root@ip-172-31-9-43 mybranchrepo]# git branch
* branch1
  master
[root@ip-172-31-9-43 mybranchrepo]# touch mybranchdata[1..3]
[root@ip-172-31-9-43 mybranchrepo]# git status
On branch branch1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        mybranchdata1
        mybranchdata2
        mybranchdata3

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 mybranchrepo]# git add .
[root@ip-172-31-9-43 mybranchrepo]# git commit -m "iam committing in my branch branch1"
[branch1 902fca5] iam committing in my branch branch1
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 mybranchdata1
create mode 100644 mybranchdata2
create mode 100644 mybranchdata3
[root@ip-172-31-9-43 mybranchrepo]# ls
mybranchdata1 mybranchdata2 mybranchdata3 test.txt
[root@ip-172-31-9-43 mybranchrepo]# git checkout master
Switched to branch 'master'
[root@ip-172-31-9-43 mybranchrepo]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:43 test.txt
[root@ip-172-31-9-43 mybranchrepo]# git branch
  branch1
* master
[root@ip-172-31-9-43 mybranchrepo]#
```

## Merging:

```
[root@ip-172-31-9-43 mybranchrepo]# git branch
  branch1
* master
[root@ip-172-31-9-43 mybranchrepo]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:43 test.txt
[root@ip-172-31-9-43 mybranchrepo]# git merge branch1
Updating 8fb80f8..902fca5
Fast-forward
 mybranchdata1 | 0
 mybranchdata2 | 0
 mybranchdata3 | 0
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 mybranchdata1
create mode 100644 mybranchdata2
create mode 100644 mybranchdata3
[root@ip-172-31-9-43 mybranchrepo]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:49 mybranchdata1
-rw-r--r-- 1 root root 0 Jul 27 18:49 mybranchdata2
-rw-r--r-- 1 root root 0 Jul 27 18:49 mybranchdata3
-rw-r--r-- 1 root root 0 Jul 27 18:43 test.txt
[root@ip-172-31-9-43 mybranchrepo]# git branch
  branch1
* master
```

## Tags:

Tags helps us to save the state of the previous things, before it created.

```
[root@ip-172-31-9-43 tmp]# mkdir mytagging
[root@ip-172-31-9-43 tmp]# cd mytagging/
[root@ip-172-31-9-43 mytagging]# ls -l
total 0
[root@ip-172-31-9-43 mytagging]# pw
[root@ip-172-31-9-43 mytagging]# pwd
/tmp/mytagging
[root@ip-172-31-9-43 mytagging]# git init .
Initialized empty Git repository in /tmp/mytagging/.git/
```

```
[root@ip-172-31-9-43 mytagging]# git init .
Initialized empty Git repository in /tmp/mytagging/.git/
[root@ip-172-31-9-43 mytagging]# touch test{1..4}.txt
[root@ip-172-31-9-43 mytagging]# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    test1.txt
    test2.txt
    test3.txt
    test4.txt

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-9-43 mytagging]# git add .
[root@ip-172-31-9-43 mytagging]# git commit -m "working with git tag"
[master (root-commit) 64f968a] working with git tag
 4 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 test1.txt
 create mode 100644 test2.txt
 create mode 100644 test3.txt
 create mode 100644 test4.txt
[root@ip-172-31-9-43 mytagging]# git tag v1.0
[root@ip-172-31-9-43 mytagging]# touch finaltest{1..4}.txt
[root@ip-172-31-9-43 mytagging]# git add .
[root@ip-172-31-9-43 mytagging]# git commit -m "testing tag option"
[master bff302b] testing tag option
 4 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 finaltest1.txt
 create mode 100644 finaltest2.txt
 create mode 100644 finaltest3.txt
 create mode 100644 finaltest4.txt
```

```
[root@ip-172-31-9-43 mytagging]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:52 finaltest1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 finaltest2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 finaltest3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 finaltest4.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test4.txt
[root@ip-172-31-9-43 mytagging]#
```

Now let's revert back to the previous state i.e where we have our files test1.txt,test2.txt,test3.txt,test4.txt

```
[root@ip-172-31-9-43 mytagging]# git tag
v1.0
[root@ip-172-31-9-43 mytagging]# git log
commit bff302b7715dbcaac36fa203bc12c95df616691e (HEAD -> master)
Author: gituserid <shahan.aix@gmail.com>
Date: Mon Jul 27 18:53:15 2020 +0000

    testing tag option

commit 64f968a81ce6b7e77ba05b279ed11898e4fe9079 (tag: v1.0)
Author: gituserid <shahan.aix@gmail.com>
Date: Mon Jul 27 18:52:29 2020 +0000

    working with git tag
```

```
working with git tag
[root@ip-172-31-9-43 mytagging]# git checkout 64f968a81ce6b7e77ba05b279ed11898e4fe9079
Note: switching to '64f968a81ce6b7e77ba05b279ed11898e4fe9079'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 64f968a working with git tag
[root@ip-172-31-9-43 mytagging]# git checkout -b mynewrecovery
Switched to a new branch 'mynewrecovery'
[root@ip-172-31-9-43 mytagging]# ls -l
total 0
-rw-r--r-- 1 root root 0 Jul 27 18:52 test1.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test2.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test3.txt
-rw-r--r-- 1 root root 0 Jul 27 18:52 test4.txt
[root@ip-172-31-9-43 mytagging]# git branch
* mynewrecovery
master
[root@ip-172-31-9-43 mytagging]#
```

## Git Log:

**git log --pretty=oneline**

```
[root@ip-172-31-9-43 mytagging]# git log
commit 64f968a81ce6b7e77ba05b279ed11898e4fe9079 (HEAD -> mynewrecovery, tag: v1.0)
Author: gituserid <shahan.aix@gmail.com>
Date: Mon Jul 27 18:52:29 2020 +0000

    working with git tag

[root@ip-172-31-9-43 mytagging]# git log --pretty=oneline
64f968a81ce6b7e77ba05b279ed11898e4fe9079 (HEAD -> mynewrecovery, tag: v1.0) working with git tag
[root@ip-172-31-9-43 mytagging]#
```

## Working with Github :

### Account Signup

Team Enterprise Explore Marketplace Pricing

Search GitHub Sign in Sign up

*Fill all the data*

**Built for developers**

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 50 million developers.

Username  
myuserid123 ✓

Email  
myid123@gmail.com ✓

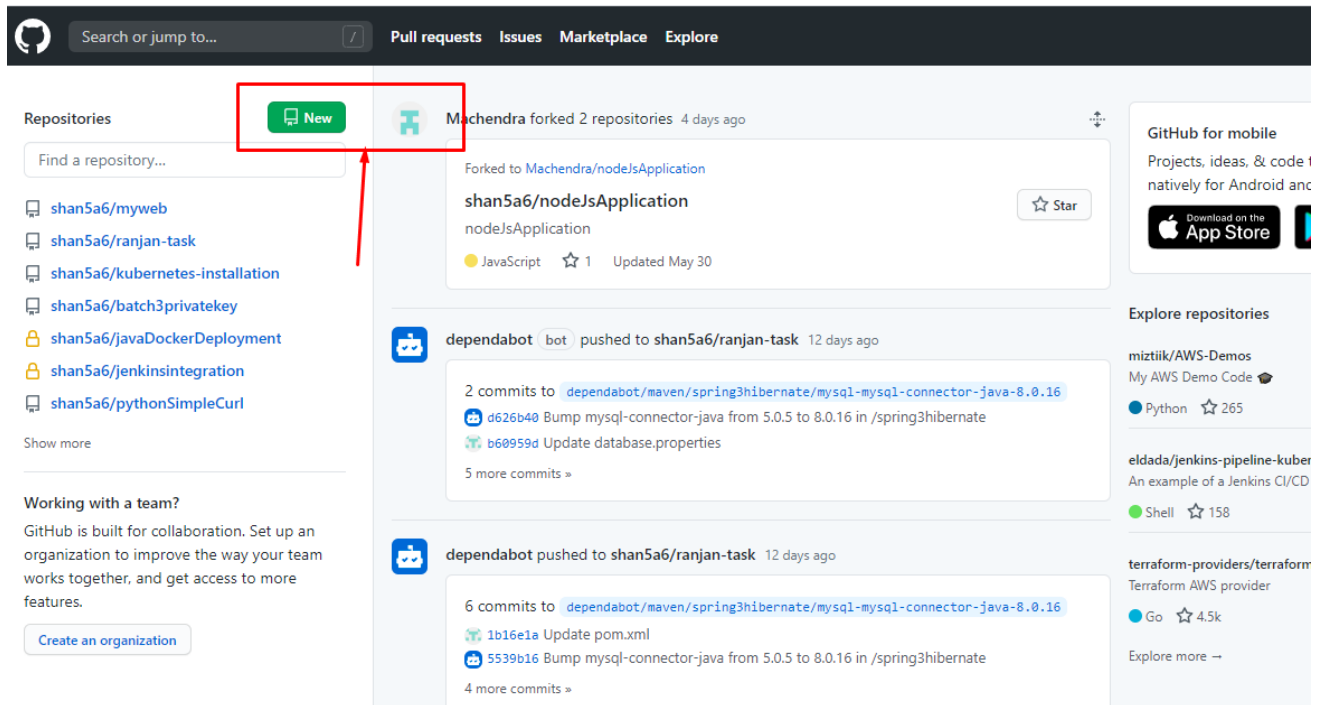
Password  
..... ✓

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

**Sign up for GitHub**

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.

## Creating a repository:




## Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \* shans5a6 / Repository name \* dvbatch4 ✓

Great repository names are short and memorable. Need inspiration? How about [turbo-chainsaw](#)?

Description (optional)

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

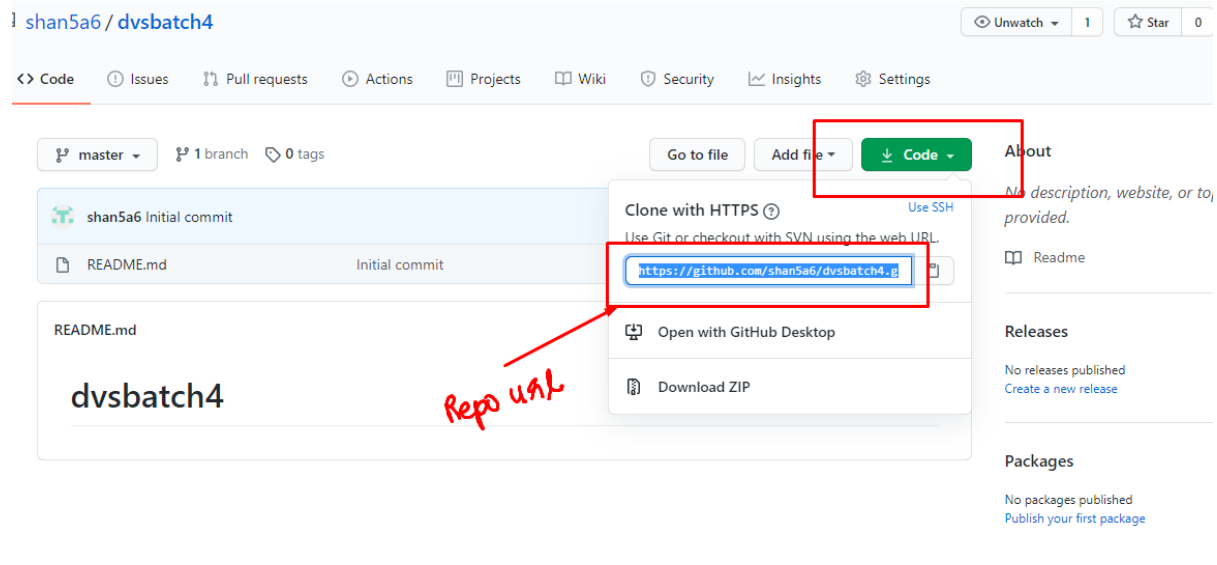
☒ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None ⓘ

**Create repository**

## Cloning Repository to server :

DVS Technologies, Opp Home Town, Beside Biryani Zone, Marathahalli, Bangalore Phone: 9632558585 Mobile: 8892499499 Mail : [dvs.training@gmail.com](mailto:dvs.training@gmail.com) Web: [www.dvstechnologies.in](http://www.dvstechnologies.in)



```
[root@gitserver tmp]# git clone https://github.com/shan5a6/dvsbatch4.git
Cloning into 'dvsbatch4'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
[root@gitserver tmp]#
```

## Pushing Data to Github under our branch :

```
[root@gitserver tmp]# git clone https://github.com/shan5a6/dvsbatch4.git
Cloning into 'dvsbatch4'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
[root@gitserver tmp]# cd dvsbatch4/
[root@gitserver dvsbatch4]# ls -l
total 4
-rw-r--r-- 1 root root 11 Jul 28 03:05 README.md
[root@gitserver dvsbatch4]#
[root@gitserver dvsbatch4]# git checkout -b mytestbranch
Switched to a new branch 'mytestbranch'
[root@gitserver dvsbatch4]# touch file{1..4}.txt
[root@gitserver dvsbatch4]# ls -l
total 4
-rw-r--r-- 1 root root 0 Jul 28 03:06 file1.txt
-rw-r--r-- 1 root root 0 Jul 28 03:06 file2.txt
-rw-r--r-- 1 root root 0 Jul 28 03:06 file3.txt
-rw-r--r-- 1 root root 0 Jul 28 03:06 file4.txt
-rw-r--r-- 1 root root 11 Jul 28 03:05 README.md
[root@gitserver dvsbatch4]# git add .
[root@gitserver dvsbatch4]# git commit -m "testing my github repo"
[mytestbranch b28f4e2] testing my github repo
4 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1.txt
create mode 100644 file2.txt
create mode 100644 file3.txt
create mode 100644 file4.txt
```



### Verifying data in the github repository:

DVS Technologies, Opp Home Town, Beside Biryani Zone, Marathahalli, Bangalore Phone: 9632558585 Mobile: 8892499499 Mail : [dvs.training@gmail.com](mailto:dvs.training@gmail.com) Web: [www.dvstechnologies.in](http://www.dvstechnologies.in)

mytestbranch had recent pushes 6 minutes ago [Compare & pull request](#)

mytestbranch 2 branches 0 tags [Go to file](#) [Add file](#) [Code](#)

This branch is 1 commit ahead of master. [Pull request](#) [Compare](#)

shan5a6 testing my github repo b28f4e2 8 minutes ago 2 commits

README.md	Initial commit	12 minutes ago
file1.txt	testing my github repo	8 minutes ago
file2.txt	testing my github repo	8 minutes ago
file3.txt	testing my github repo	8 minutes ago
file4.txt	testing my github repo	8 minutes ago

## About

No description, website, or topics provided.

[Readme](#)

## Releases

No releases published  
[Create a new release](#)

## Packages

No packages published  
[Publish your first package](#)

## Raising a pull request :

shan5a6 / dvsbatch4 [Unwatch](#) 1 [Star](#) 0 [Fork](#)

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

mytestbranch had recent pushes 6 minutes ago [Compare & pull request](#)

mytestbranch 2 branches 0 tags [Go to file](#) [Add file](#) [Code](#)

This branch is 1 commit ahead of master. [Pull request](#) [Compare](#)

shan5a6 testing my github repo b28f4e2 10 minutes ago 2 commits

README.md	Initial commit	14 minutes ago
file1.txt	testing my github repo	10 minutes ago
file2.txt	testing my github repo	10 minutes ago
file3.txt	testing my github repo	10 minutes ago
file4.txt	testing my github repo	10 minutes ago

<https://github.com/shan5a6/dvsbatch4/compare/mytestbranch?expand=1>

## About

No description, website, or topics provided.

[Readme](#)

## Releases

No releases published  
[Create a new release](#)

## Packages

No packages published  
[Publish your first package](#)

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base: master ← compare: mytestbranch ✓ Able to merge. These branches can be automatically merged.

testing my github repo

Write Preview H B I ≡ <> @ ↩

write the changes and description for this pull request

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

## Merging my PR(Pull request):

### testing my github repo #1

Open shan5a6 wants to merge 1 commit into master from mytestbranch

Conversation 0 Commits 1 Checks 0 Files changed 4 +0 -0

shan5a6 commented now

write the changes and description for this pull request

testing my github repo b28f4e2

Add more commits by pushing to the mytestbranch branch on shan5a6/dvsbatch4.

Continuous integration has not been set up

GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

✓ This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request You can also open this in GitHub Desktop or view command line instructions.

Reviewers

No reviews

Still in progress? Convert to draft

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

DVS Technologies, Opp Home Town, Beside Biryani Zone, Marathahalli, Bangalore Phone: 9632558585 Mobile: 8892499499 Mail : [dvs.training@gmail.com](mailto:dvs.training@gmail.com) Web: [www.dvstechnologies.in](http://www.dvstechnologies.in)

## testing my github repo #1

**Open** shan5a6 wants to merge 1 commit into `master` from `mytestbranch`

Conversation 0 Commits 1 Checks 0 Files changed 4



shan5a6 commented 1 minute ago

Owner

write the changes and description for this pull request

testing my github repo

b28f4e2

Add more commits by pushing to the `mytestbranch` branch on shan5a6/dvsbatch4.



Merge pull request #1 from shan5a6/mytestbranch

testing my github repo

Confirm merge

Cancel

Reviewers

No reviews

Still in progress? Conver

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

## testing my github repo #1

**Merged** shan5a6 merged 1 commit into `master` from `mytestbranch` now

Conversation 0 Commits 1 Checks 0 Files changed 4



shan5a6 commented 2 minutes ago

Owner

write the changes and description for this pull request

testing my github repo

b28f4e2



shan5a6 merged commit 3a26c33 into `master` now

Revert



Pull request successfully merged and closed

Delete branch

shan5a6 / dvsbatch4

Unwatch 1 Star 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 2 branches 0 tags

Go to file Add file Code

shan5a6 Merge pull request #1 from shan5a6/mytestbranch 3a26c33 29 seconds ago 3 commits

README.md	Initial commit	19 minutes ago
file1.txt	testing my github repo	15 minutes ago
file2.txt	testing my github repo	15 minutes ago
file3.txt	testing my github repo	15 minutes ago
file4.txt	testing my github repo	15 minutes ago

README.md

dvsbatch4

About

No description, website, or topics provided.

Readme

Releases

No releases published  
[Create a new release](#)

Packages

No packages published  
[Publish your first package](#)

## Making Private & Deleting the repository:

shan5a6 / dvsbatch4

Unwatch 1 Star 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 2 branches 0 tags

Go to file Add file Code

shan5a6 Merge pull request #1 from shan5a6/mytestbranch 3a26c33 6 minutes ago 3 commits

README.md	Initial commit	25 minutes ago
file1.txt	testing my github repo	21 minutes ago
file2.txt	testing my github repo	21 minutes ago
file3.txt	testing my github repo	21 minutes ago
file4.txt	testing my github repo	21 minutes ago

README.md

dvsbatch4

About

No description, website, or topics provided.

Readme

Releases

No releases published  
[Create a new release](#)

Packages

No packages published  
[Publish your first package](#)

## Scroll down the site

### Danger Zone

**Change repository visibility**  
This repository is currently public.

**Change visibility**

**Transfer ownership**  
Transfer this repository to another user or to an organization where you have the ability to create repositories.

**Transfer**

**Archive this repository**  
Mark this repository as archived and read-only.

**Archive this repository**

**Delete this repository**  
Once you delete a repository, there is no going back. Please be certain.

**Delete this repository**

**Danger Zone**

**Change repository visibility**  
This repository is currently public.

☒ **Make public**  
This repository is currently public.

☐ **Make private**  
Hide this repository from the public.

Please type shan5a6/dvsbatch4 to confirm.

**I understand, change repository visibility.**

**Change visibility**

**Transfer**

**Archive this repository**

**Delete this repository**

- Multiple assignees for issues and PRs
- Multiple reviewers for PRs
- Branch protection rules.

Please type `shan5a6/dvsbatch4` to confirm.

shan5a6/dvsbatch4

**I understand, change repository visibility.**

Confirm password to continue

Password

[Forgot password?](#)

**Confirm password**

Tip: You are entering [sudo mode](#). We won't ask for your password again for a few hours.

## Cloning git repo with ssh keys:

## Generating keys:

**You can use "puttygen" or use the server "ssh-keygen"**

```
[root@git tmp]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Pn8pKq1FY/8hLMelFIUfbFKwOvrLnLdFKjdESh5vsN8 root@git
The key's randomart image is:
+---[RSA 2048]----+
|          .          |
|        + .o         |
|       o B. .         |
|      S.++.          |
|     +oOo+=          |
|    o*.@=Eo          |
|   o=.X+**+.         |
|  .+X=++.            |
+----[SHA256]-----+
[root@git tmp]#
```

**Public key:**

```
[root@git tmp]# cat /root/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAQXj0YDejL4ZY++Y26EPp/MtnwV18tJThqXGU//Yn64ww6AJwMlXxnnSwrODSPyAdu
hJp4bj5DGe55Q387jphobZ0nsrCgC7ckbb2mnxTRZKraTro/wUBBP6drZ2xrfgyb1NEf6K5Csw7NtrbeMAYsudecNn8a/FK0KHfQgyJ
kUOK1RkzU+PPMcIQaFuBR7Ufkzgy4JMtPppYc4Sgn9mvYrMXq/opsiOBGfi7pAYWpScPinJBXOVxRkCbl63AR0D33Pry8Ueg0DzZpIOE0
Hae9R8r2t+kSnFbrePhwvEpJm7t+8Bre0Si8JwZn+1NGQHXn1kRUGOzjUabmvedLDwg root@git
```

**Private key:**

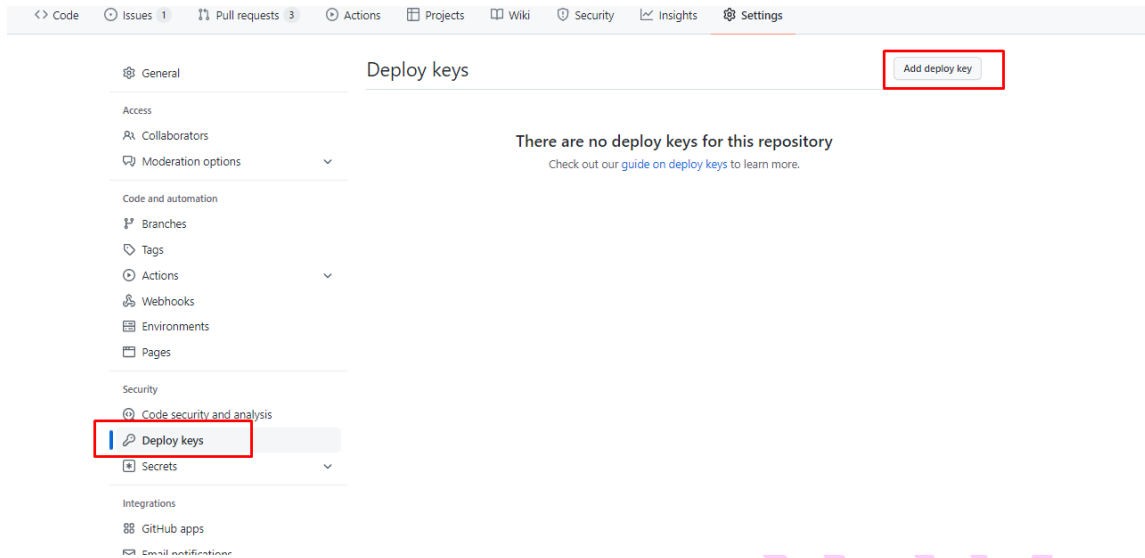
```

root@kali:~# cat /root/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAA19mg3oy+GWFvmNuhD6zfLZ8FdfLSU4a1xlp/2J+uMMOGcCd
jCv550ak2zqoWAHh0SaeG4+QxnuUDeat+4y6A2cd7KwAoA3JG29pp8SEWSzLa6F
8AVAT+na2dsq34L2t9TRht+igtQs03DUb2t3jAGLNNdTFGv3ytcHk30K1ZFd1tZRP
jzzHCeGhgbyGuE1H5M4MuCTLbT6aWHOEOj/2r2Kzf6v6kbJgJr3y86QM1j3EoLTsqV
zU5NA2zetwEt9z2MvfH0NA82aSNBNB2nvUSKwdfPEx+kW63jxY6vRD4yZu//ga
3qGoeVCszf7tTRK8BTz3L1Bj5s4lGm5r3n5w8khwIDAQOABAIBAQcJK6UGjVR9Mcn
nrrFLk22thtndHmHjVswH/rAwQJU2NOjgYfpypp8DQ0Z0B7yQ2M4he1ozLEfhj
laxalJyEi95lms78TzYfv4Nh7H8mpAqyucX2MZBqLadtCm7ZzG2A2rYntYdMnZNfH
P0T2Z3C/LYOG1+HcdBOTrfivucp0wTnmTnt8VLo/atht12gI9c9PctjLlQsgs
J5u7ZKqdmboJkeoFYtCxiQY0vjJfs/B522bgHkHbsGQCfGLS85qreaGdKHkko+yiUw
cMMKkgb1WN9S4r4tREMSt1oov7+xaAQ4OHhJ+HsQV1BHVf5S2tLebmA0wUuqJp5c
4UyGnSvzxAoGAAPvKsUGBkKaEC+nylfggW654DX0014snsU0t1vYs1YQl4YAD4X
+Jom3qAKUCFq75UEgdcE8CdmzrnF5drYlARAX95G5GfjTHlWku1tW0c6fQ0204
Jy0qz32Rq4H9r0VzZfEALM9K1XsgHwXqEzH6X9K0e0BrRtu74a7AoGBANSt
Wp1z4g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1g1
3vjf4g8sXOKZDpzwuY5s9qG8eSOlPq3X1k780V9b3WF4M/TPb45A0p9r9W9L
hm5lFcpSf6jGqRqemfR1NAMFodQIBYlYHBPBgIRPaOGA6BMKM74/YfyzB5e1shgLB
GraulrGy10ueRQOKIMxHY4AlXFFOT68UFz4l9AQz3cm0oS+0Moho6QM1ak4t9Xyc
tFAPUL0tAAezqKbWLnQ1yTC6gnuoYUuNg767pQOSGHdX4lSZN+atKO5a90s+NOxkfT
C4DelylJ4Hm29833x/H/ly5nRAoGAfGwLDeL8x1jGHAS5gPygGevpau6egUB05dN
16vMIGc0Augc6d+Yby6bd4voqDvBzJqgDbhOWPqKXKE21vbesaurJlJPbfng7VAX
3gv0y2Sug912R1AVDKMf4DXHAvnJr3jnuNuH7jKwLoIA01a2DsttTV/Bj3pxHleSH
Y0V/jPUCGVEApnBz67DwokrLLku7KH5fbA2Rlchbo-r83MKuA9az6fZ12NqrwFk8b
bHVGtRn6bGwrg607z8Elq8EO8AhHgvnTrPlcmABCFPy18L+J4/lbTNPLTSLxjgm
LnhEkus1h1Bjci434AQ0C1NO6bw85GzLU214T8AVKSo299E1Y4B5Eqk=
-----END RSA PRIVATE KEY-----

```

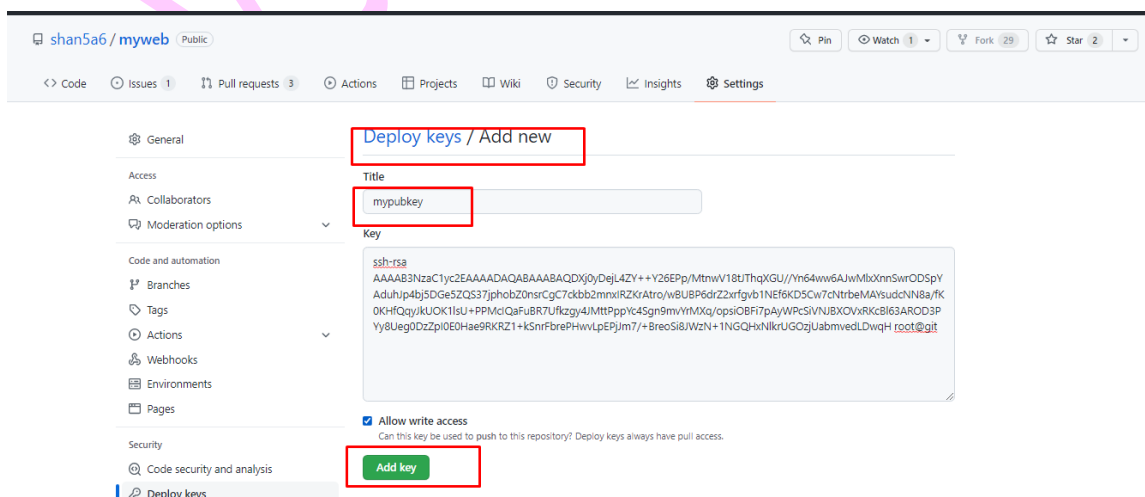
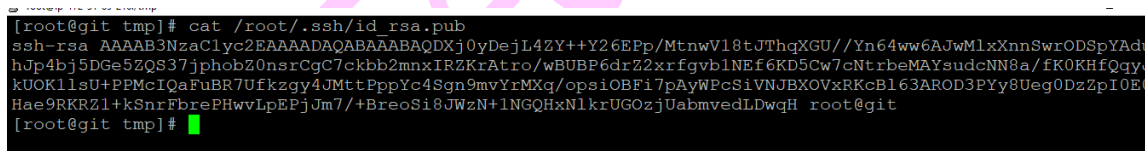


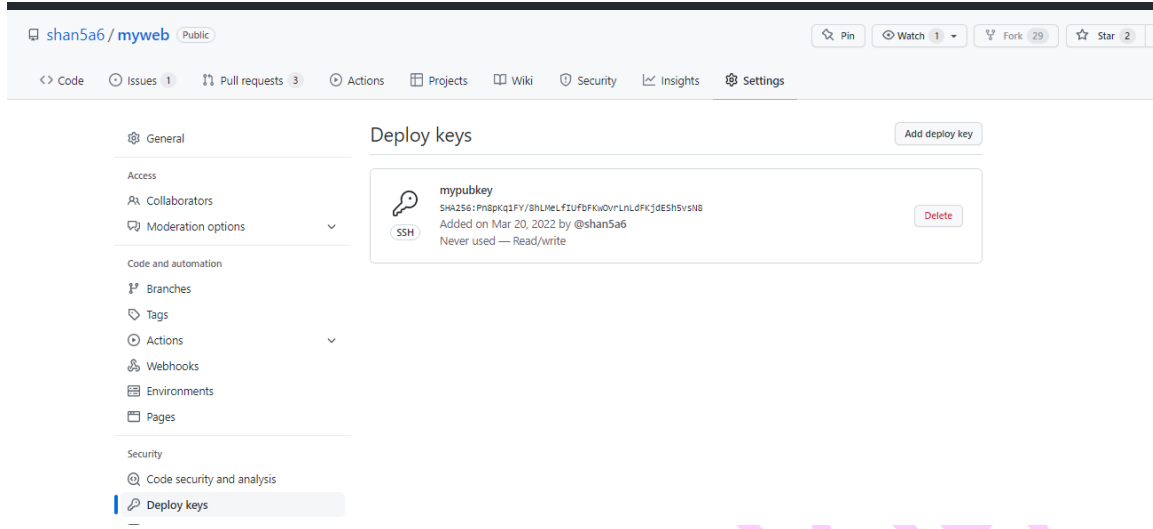
## Configure publickey in the github:



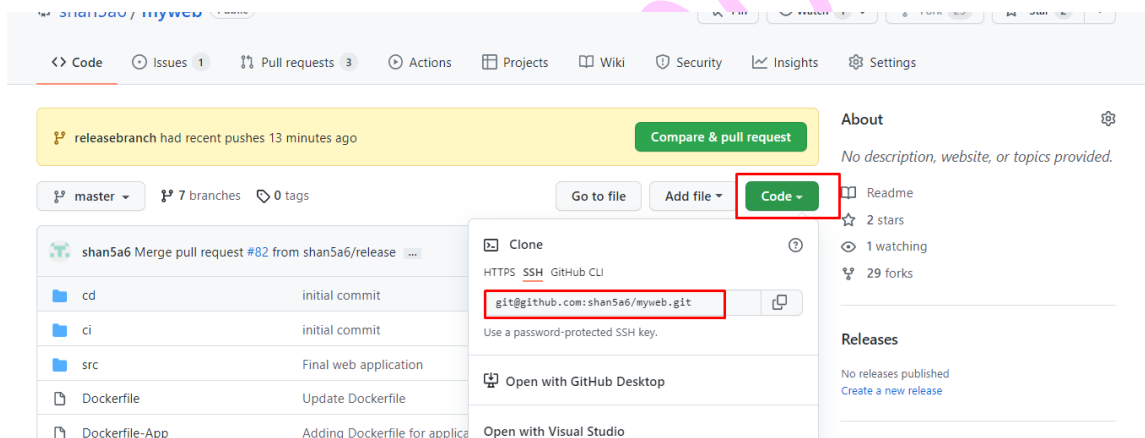
### ssh-rsa

```
AAAAB3NzaC1yc2EAAAADAQABAAQADXDj0yDejL4ZY++Y26EPp/MtnwV18tJThqXGU//Yn64ww6AJwMlxXnnSwrODSpYAduhJp4bj5DGe5ZQS37jphobZ0nsrCgC7ckbb2mnxIRZKrAtro/wBUBP6drZ2xrfgvb1NEf6KD5Cw7cNtrbeMAYsudcNN8a/fK0KHfQqyJkUOK1lsU+PPMcIQaFuBR7Ufkzgy4JMtPppYc4Sgn9mvYrMXq/opsiOBFi7pAyWPcSiVNBXOVxRKcBl63AROD3PYy8Ueg0DzZpI0E0Hae9RKRZ1+kSnrFbrePHwvLpEPjJm7/+BreoSi8JWzN+1NGQHxNlkrUGOzjUabmvedLDwqH root@git
```





Now clone the repository with ssh:



```
root@ip-172-31-89-216/tmp/myweb
[root@git tmp]# git clone git@github.com:shan5a6/myweb.git
Cloning into 'myweb'...
The authenticity of host 'github.com (140.82.114.3)' can't be established.
ECDSA key fingerprint is SHA256:p2QAMXNIC1TJYWeI0trVc98/R1BUFWu3/LiyKgUfQM.
ECDSA key fingerprint is MD5:7b:99:81:1e:4c:91:a5:0d:5a:2e:2e:80:13:3f:24:ca.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,140.82.114.3' (ECDSA) to the list of known hosts.
remote: Enumerating objects: 400, done.
remote: Counting objects: 100% (142/142), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 400 (delta 99), reused 78 (delta 67), pack-reused 258
Receiving objects: 100% (400/400), 5.90 MiB | 31.00 MiB/s, done.
Resolving deltas: 100% (125/125), done.
[root@git tmp]# ls -l myweb/
total 40
-rw-r--r-- 1 root root 544 Mar 20 12:07 ansible.cfg
drwxr-xr-x 2 root root 25 Mar 20 12:07 cd
drwxr-xr-x 2 root root 25 Mar 20 12:07 ci
-rw-r--r-- 1 root root 1096 Mar 20 12:07 context.xml
-rw-r--r-- 1 root root 210 Mar 20 12:07 Dockerfile
-rw-r--r-- 1 root root 196 Mar 20 12:07 Dockerfile-App
-rw-r--r-- 1 root root 1181 Mar 20 12:07 Dockerfile-Base
-rw-r--r-- 1 root root 17 Mar 20 12:07 hosts
-rw-r--r-- 1 root root 1260 Mar 20 12:07 myweb.yml
-rw-r--r-- 1 root root 2758 Mar 20 12:07 pom.xml
-rw-r--r-- 1 root root 10 Mar 20 12:07 README.md
drwxr-xr-x 4 root root 30 Mar 20 12:07 src
-rw-r--r-- 1 root root 0 Mar 20 12:07 test1
-rw-r--r-- 1 root root 391 Mar 20 12:07 tomcat-users.xml
[root@git tmp]# cd myweb/
[root@git myweb]# git remote -v
```

```
[root@git myweb]# git remote -v
origin  git@github.com:shan5a6/myweb.git (fetch)
origin  git@github.com:shan5a6/myweb.git (push)
[root@git myweb]#
```