

Gitlab Configuration

`git init` ← initialize project directory

`git config --global user.name "vijayendra.rathod@gmail.com"` ← configure username

`git remote add origin https://gitlab.com/vijayendra_rathod/myrepo.git` ← add remote path

`git remote add .` [add all file and folder to staging area]

`git commit -m "first commit"`

`git push -uf origin main` ← push the code to branch

Create | Delete | Push Branch

create local branch

`git branch mybranch`

list branch

`git branch --list`

```
main
* master
mybranch
```

* means Active working Branch

Delete Remote branch

`git push -d origin mybranch`

Delete Local branch

`git branch -d mybranch`

Change the Branch

`git switch branch`

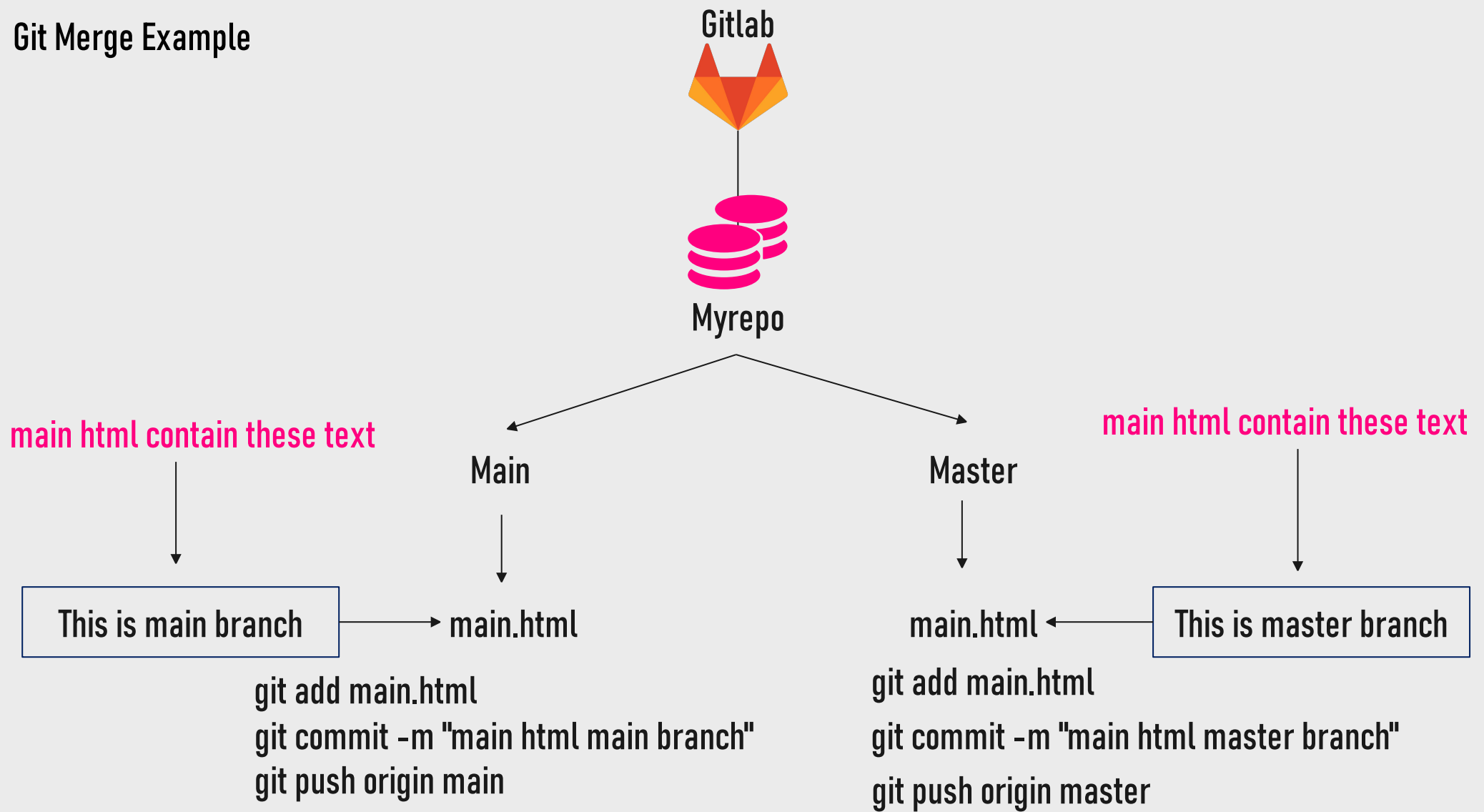
Push to branch

`git push origin main`

Delete Remote branch

`git push -d origin mybranch`

Git Merge Example



`git checkout main`

Git Merge | conflicts Resolve

git merge master

```
$ git merge master
CONFLICT (add/add): Merge conflict in main.html
Auto-merging main.html
Automatic merge failed; fix conflicts and then commit the result
```

```
Sam@DESKTOP-NI9KMNN MINGW64 ~/Desktop/app-main (main|MERGING)
$ vi main.html
```

```
+<<<<<< HEAD
| this is main branch
|
|=====
|+this is Master branch
|+>>>>>> master
```

save the file


```
this is main branch
this is Master branch
```



```
Sam@DESKTOP-NI9KMNN MINGW64 ~/Desktop/app-main (main|MERGING)
$ git commit -a -m "merge conflicts resolved"
[main 08a8588] merge conflicts resolved
```

```
Sam@DESKTOP-NI9KMNN MINGW64 ~/Desktop/app-main (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 332 bytes | 332.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/vijayendra_rathod/myrepo.git
484a2fd..08a8588 main -> main
```

Merge Complete

main ▾ myrepo / m

 merge conflicts resolved
vijayendra.rathod@gmail.com

 main.html  45 bytes

1	
2	this is main branch
3	this is Master branch

Git Merge | conflicts Resolve

git merge master

```
$ git merge master
CONFLICT (add/add): Merge conflict in main.html
Auto-merging main.html
Automatic merge failed; fix conflicts and then commit the result
```

```
Sam@DESKTOP-NI9KMNN MINGW64 ~/Desktop/app-main (main|MERGING)
$ vi main.html
```

```
+<<<<<< HEAD
| this is main branch
+=====
+this is Master branch
+>>>>>> master
```

save the file

```
this is main branch
this is Master branch
```


```
Sam@DESKTOP-NI9KMNN MINGW64 ~/Desktop/app-main (main|MERGING)
$ git commit -a -m "merge conflicts resolved"
[main 08a8588] merge conflicts resolved
```



```
Sam@DESKTOP-NI9KMNN MINGW64 ~/Desktop/app-main (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 332 bytes | 332.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/vijayendra_rathod/myrepo.git
484a2fd..08a8588 main -> main
```

Merge Complete

main

myrepo / m

 merge conflicts resolved
vijayendra.rathod@gmail.com

 main.html  45 bytes

1	
2	this is main branch
3	this is Master branch

Configuring Gitlab Runner on Linux

1



settings >> CI/CD >> Runner

These runners are specific to this project.

Set up a specific runner for a project

1. Install GitLab Runner and ensure it's running.
2. Register the runner with this URL:

<https://gitlab.com/>

And this registration token:

GR1348941jBi7StCxgdv2hQmHpTpY

Reset registration token

Show runner installation instructions

Available specific runners

● #14436919 (T_LpU-y-) 🔒

✎ || Remove runner

My Runner

linux

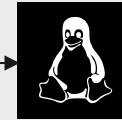
shell executor

3

```
sudo gitlab-runner register \  
--url https://gitlab.com/ \  
--registration-token $REGISTRATION_TOKEN \  
--name My Runner \  
--executor shell
```

this will configure shell executor

4



Ubuntu 20.0.4

2

This will install gitlab-runner

```
sudo curl -L --output /usr/local/bin/gitlab-runner  
https://gitlab-runner-downloads.s3.amazonaws.com/  
latest/binaries/gitlab-runner-linux-amd64
```

```
# Give it permission to execute  
sudo chmod +x /usr/local/bin/gitlab-runner
```

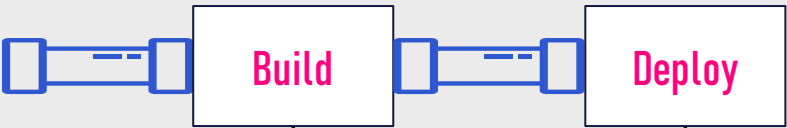
```
# Create a GitLab Runner user  
sudo useradd --comment 'GitLab Runner' --create-home  
gitlab-runner --shell /bin/bash
```

```
# Install and run as a service  
sudo gitlab-runner install --user=gitlab-runner --  
working-directory=/home/gitlab-runner  
sudo gitlab-runner start
```

```
usermod -aG docker gitlab-runner
```

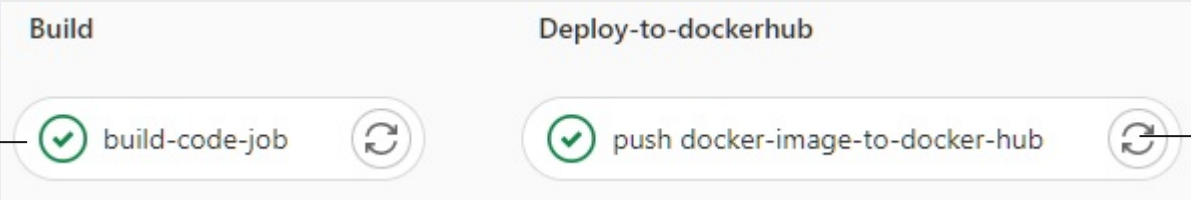
install docker engine on system

Execute pipeline



Docker image from Code

Deploy Docker Image to DockerHub



```
1 Removing intermediate container 0b2e38ab8f1c
2 ---> eafd73146f80
3 Step 6/7 : CMD ["node", "src/index.js"]
4 ---> Running in a23b6fcafea2
5 Removing intermediate container a23b6fcafea2
6 ---> e76cda3fa2cb
7 Step 7/7 : EXPOSE 3000
8 ---> Running in 446d5d32bb7a
9 Removing intermediate container 446d5d32bb7a
0 ---> 0b49eelf3f78
1 Successfully built 0b49eelf3f78
2 Successfully tagged vijayendra99/my-app:latest
4 Cleaning up project directory and file based variables
6 Job succeeded
```

```
$ docker push vijayendra99/my-app
Using default tag: latest
The push refers to repository [docker.io/vijayendra99/my-app]
cf7223c0e39f: Preparing
cd9d8c24d2f8: Preparing
805f4cccdfe: Preparing
b7b6af48ba71: Preparing
78df64f39f97: Preparing
d08e1bbcb0ea: Preparing
33d708a6cd80: Preparing
alc01e366b99: Preparing
d08e1bbcb0ea: Waiting
33d708a6cd80: Waiting
alc01e366b99: Waiting
78df64f39f97: Layer already exists
d08e1bbcb0ea: Layer already exists
805f4cccdfe: Pushed
33d708a6cd80: Layer already exists
alc01e366b99: Layer already exists
cd9d8c24d2f8: Pushed
cf7223c0e39f: Pushed
b7b6af48ba71: Pushed
latest: digest: sha256:bb8d41423ed856fb728fb01eb2f961c0f386056
e: 2000
Cleaning up project directory and file based variables
Job succeeded
```