
GIT

Pushing the maven project available in local system to remote repo.

---

1. Go inside the maven webapp folder

2. Open the git bash terminal ---> git init

3. Add the files to the staging area ---> git add .

4. Commit the files to the local repository ---> git commit -m "<Commit Message>"

5. Goto github and create a public repository

6. git branch -M main

7. Add the remote repository ---> git remote add origin git@github.com:KastroVKiran/SA-Maven-Web-App-Kastro.git

8. Push the code ---> git push -u origin main

GIT STASH

---

1. Clone a github repo (Ex: maven webapp)

2. Add a dependency in pom.xml file (or) Do some change in the pom.xml

3. git status (pom.xml will be in red colour, because it is not added to staging area and commited)

Assume you are in now in the middle of the work. You have to keep this work aside (store it in temporary location) and finish the newly assigned task.

4. To store the present work in a temporary location

git stash

5. Goto pom.xml file, you dont see the dependency added in Step 2. This is now saved in temporary location.

6. Begin the new task (Creation of a file)

touch kastro.py ---> Add the file ---> Commit the file

The new task is finished

7. To go back to the previous work;

git stash apply

You can see the dependency added in Step 2

Now you add this to staging area and commit

8. To see the list of stashs

git stash list

You can see the "stash id" also.

Note: To get back the specific file from stash zone;

git stash apply <StashID>

GIT BRANCHES

=> To maintain a separate code base for each team, we will work with branches concept.

=> In real-time, DevOps engineers have to create the branches.

=> By default, MASTER branch will get created.

=> MAIN branch is also available

Netflix

1. Movies - Dev1 - movies related code/files
2. Documentaries - Dev2 - documentaries related code/files
3. Sports - Dev3 - sports related code/files
4. Cartoons - Dev4 - cartoons related code/files

To create a branch

git branch <BranchName>

To see the list of branches

git branch

To switch the branches

git checkout <BranchName>

Note: When we are creating a new branch from any other branch, whatever files are available in the other branch, all those files will automatically copied to the new branch

To push the files and automatically create a branch in the github

git push origin <BranchName>

Note: if you are unable to push (or) if the git terminal is asking for username and password, we have to create a Personal Access Token (PAT)

Git Ignore is used to not add and commit the files

To ignore the files, we will create a file called .gitignore and then we will add all the files to the .gitignore whichever we want to ignore to add and commit.

Note: Files in .gitignore cannot be seen in github repository.

To rename a branch

Renaming the branches can be done in 2 ways

In git terminal: git branch -m <OldBranchName> <NewBranchName>

In github account

Note: When you rename branch in gitterminal, the branch name will not get changed in the github repo.

Pull Request

If you want to get the files from one branch to another branch we will use pull request concept.

Using Pull request we can merge the branches which is known as branch merging.

Deleting a branch

---

git branch -D <BranchName>

By
Kastro Kiran V

LinkedIn: https://www.linkedin.com/in/kastro-kiran/

YouTube: https://www.youtube.com/playlist?list=PLs-PsDpuAuTdOcZa-DDgG8KRbtMI_XRrC