GitHub for working commands

Username: Dhanamjaya

Password: Abc@11@abc

Provide credintials :--

git config --global user.name "Dhanamjaya"

git config --global user.email "dhanamjaya.k567@gmail.com"

Check : git config --list

STS Working :-

Goto🡪windows🡪showview 🡪search git🡪choose git repository and git staging .

STS Working History :-

Goto 🡪Git Repository🡪WorkingTree🡪right click on workingtree🡪showin(Alt+shift+W)🡪choose history

Untracked files: newly created files

Un-Staged files: Modified files

Staged files: which are ready to commit

Step 1 : git init ☹(for local repository click on (view-hidden file))

It is used to make an existing project as a Git project.

Ex: .git folder is created in local

Step 2 : git status :

list all the untracked files

Step 3 : git add filename.txt (all files)-> or git add -A or git add . or git add \*.java or  git add --a

It adds files to the staging area.

Step 4 : git rm \*

Remove file from stage area(for untracked)

Step : git rm –cached filename.java

Step : git rm –cached \*

Step 5: git commit -m ‘Commit message.’

Msg :Needs to provide story number with message

Step 6: git remote add origin giturl

It is used to link between local(.git)->central for connection

Step 7: git push -u origin master

First time only

Step : git push

Step 6:  git clone -b <Branch name><Repository URL>

Step :Git clone repourl

Step 7: git pull

Step 8: git restore Employee.java

Step :git reset HEAD Filename.java

Ex: if above file modified(changed) in local ,if don’t want central,we can reset back workspace

Step 9 :git checkout – test.java

Step :git checkout – \*

What ever you have done,those changes are removed from file to normal change.pls ignore local changes(discard).

Step 10: git log

To check commit history

Step 11: git pull

To take the latest code(update code)4\71 from repository(central) \*\*\*\*\*\*\*\*\*\*

Step 12 : git stash\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Scenario:

Your TL assigned a Task(HIS-122) to you mrng @8 AM and then you started working on that task.few changes already you made in few files but u didn’t commited because task is not yet completed.

Around 12:15 PM your TL told that Park HIS-122 for now and start working on HIS-134 it is more priority today please complte it by EOD.After HIS-134 is complted then start working with HIS-122

Step a:git stash when we enter this command Task: HIS-122 saved in temp memory

For Task:HIS-134 is completed then

Step b: git stash apply

Step : git stash apply stash@{0}

Now again check

git status ,now you can find

For Task: HIS-122 previous changes shows

If you want to working more git stash

Use below :

git stash list theire find list of temps(stash)

git stash clear to clear all stash \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Danger cmd\*\*\*\*\*\*

Step : git stash show -p stash@{0}:

To show the stash detatils

Step 13 :git clone –b <branch-name> <repo-url>

If we want to clone from develop branch then we used above command.if we not provide git branch name by default it takes from master branch.

Step a: : git branch

To check for which branch you are here .

Step 14 : No conflicts

To merge code from feature to develop branch 🡪pull request

Step 15 :

Conflicts :

Two developer are working on Sprint 7.3

Dhanamjaya and harish are he developers

Dhanamjaya working on HIS-300 story

Harish working on HIS-301 story

Note :latest code available in develop branch

Master🡪develop(sprint 7.3)🡪1. Dhanamjaya –feature-300

2. Harish--------feature-301

Step 1:

Step 1:

Step 1:

Step 1:

Step A :git checkout -b <new-branch>

Create new local branch

Step B : git push --set-upstream origin feature

Directly send code from local(.git) to central repository for other branch(feature)

Step D :git branch –list

To check list of branches available

Step C : git checkout <branch>

To change branch name

step 1:git diff

red color:old data

green color--for Workspace area new changes(workspace-stage)

note :intail empty

step 2:git diff --cached

green color-- for staging area new changes(before commit)( (workspace-stage area)

step:git commit –m ‘message’

after commit ,again enter step 2 cmd, it shows empty

step3:git diff head

clearly difference between local and(.git(local)-workspace)

Note :github credintials

Goto🡪controlpanel🡪User accounts🡪Manage windows credintials🡪 check git credintials 🡪remove

>Git help ------------🡪here available git commands all

>git help <cmd name>🡪

Ex: git help init