

WEEK 8

1. Git-HOL

Step 1: Setup your machine with Git Configuration

To create a new repository, signup with GitLab and register your credentials

Login to GitLab and create a "GitDemo" project

1. To check if Git client is installed properly: Open Git bash shell and execute

```
$ git version  
git version 2.21.0.windows.1
```

If output shows Git with its version information that indicates, that Git Client installs properly.

2. To configure user level configuration of user ID and email ID execute

```
$ git config --global user.name "username"  
$ git config --global user.email "username@cognizant.com"
```

3. To check if the configuration is properly set, execute the following command.

```
$ git config --global --list  
user.name=username  
user.email=username@cognizant.com
```

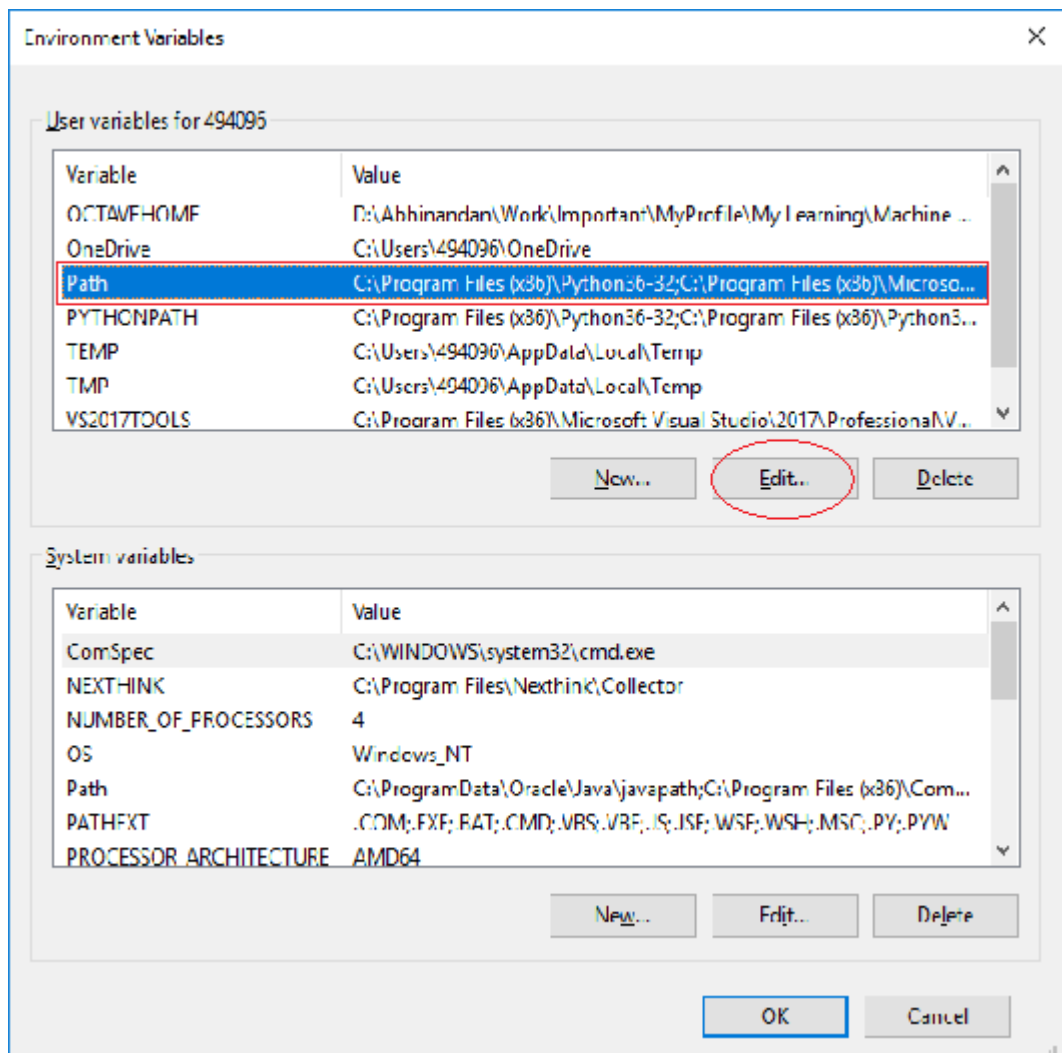
Step 2: Integrate notepad++.exe to Git and make it a default editor

1. To check, if notepad++.exe execute from Git bash

```
$ notepad++  
bash: notepad++: command not found
```

If Git bash could not able to recognize notepad++ command that implies notepad++.exe is not added to the environment path variable.

To add path of notepad++.exe to environment variable, go to control panel -> System -> Advanced System settings. Go to Advanced tab -> Environment variables -> Add path of notepad++.exe to the path user variable by clicking on "Edit"



2. Exit Git bash shell, open bash shell and execute

```
$ notepad++
```

Now, notepad++ will open from Git bash shell

3. To create an alias command for notepad++.exe, execute

```
$ notepad++.exe bash -profile
```

Step 3: Add a file to source code repository

1. Open Git bash shell and create a new project "**GitDemo**" by executing the command

```
$ git init GitDemo
Initialized empty Git repository in D:/Development_Avecto/GitDemo/.git/
```

2. Git bash initializes the "**GitDemo**" repository. To verify, execute the command

```
$ ls -al
total 8
drwxr-xr-x 1          1049089 0 Jan 13 11:54 ./
drwxr-xr-x 1          1049089 0 Jan 13 11:54 ../
drwxr-xr-x 1          1049089 0 Jan 13 11:54 .git/
```

It will display all the hidden files in the Git “working directory”.

3. To create a file “**welcome.txt**” and add content to the file, execute the command

```
$ echo "Welcome to the version control" >> welcome.txt
```

4. To verify if the file “welcome.txt” is created, execute

```
$ ls -al
total 9
drwxr-xr-x 1 494096 1049089 0 Jan 13 12:02 ./
drwxr-xr-x 1 494096 1049089 0 Jan 13 11:54 ../
drwxr-xr-x 1 494096 1049089 0 Jan 13 12:01 .git/
-rw-r--r-- 1 494096 1049089 31 Jan 13 12:02 welcome.txt
```

5. To verify the content, execute the command

```
$ cat welcome.txt
Welcome to the version control
```

6. Check the status by executing

```
$ git status
On branch master

No commits yet

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

    welcome.txt
```

Now the file “**welcome.txt**” is available in Git “working directory”

7. To make the file to be tracked by Git repository, execute the command

```
$ git add welcome.txt
warning: LF will be replaced by CRLF in welcome.txt.
The file will have its original line endings in your working directory
```

8. To add multi line comments, we are opening default editor to comment. Execute the command

```
$ git commit
```

Notepad++ editor will open and to add multi-line comment with default editor

9. To check if local and “Working Directory” git repository are same, execute git status

```
$ git status
On branch master
nothing to commit, working tree clean
```

welcome.txt is added to the local repository.

10. Signup with GitLab and create a remote repository “**GitDemo**”

11. To pull the remote repository, execute

```
git pull origin master
```

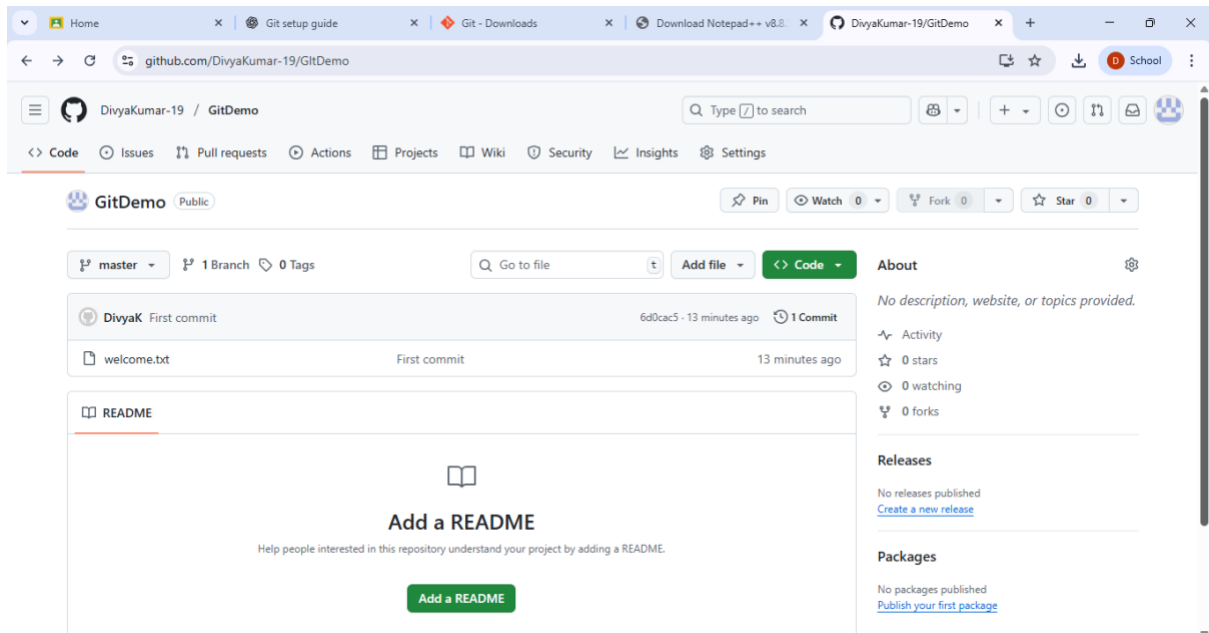
12. To push the local to remote repository, execute

```
git push origin master
```

OUTPUT:

```
MINGW64/c/Users/Divya/GitDemo
Divya@LAP19 MINGW64 ~
$ git config --global core.editor "notepad++.exe"
Divya@LAP19 MINGW64 ~
$ git config --global core.editor
notepad++.exe
Divya@LAP19 MINGW64 ~
$ mkdir GitDemo
Divya@LAP19 MINGW64 ~
$ cd GitDemo
Divya@LAP19 MINGW64 ~/GitDemo
$ git init
Initialized empty Git repository in C:/Users/Divya/GitDemo/.git/
Divya@LAP19 MINGW64 ~/GitDemo (master)
$ echo "Welcome to Git">welcome.txt
Divya@LAP19 MINGW64 ~/GitDemo (master)
$ ls
welcome.txt
Divya@LAP19 MINGW64 ~/GitDemo (master)
$ git add welcome.txt
warning: in the working copy of 'welcome.txt', LF will be replaced by CRLF the next time Git t
ouches it
Divya@LAP19 MINGW64 ~/GitDemo (master)
$ git commit -m "First commit"
[master (root-commit) 6d0cac5] First commit
1 file changed, 1 insertion(+)
create mode 100644 welcome.txt
Divya@LAP19 MINGW64 ~/GitDemo (master)
$ git remote add origin http://github.com/DivyaKumar-19/GitDemo.git
Divya@LAP19 MINGW64 ~/GitDemo (master)
$ git push -u origin master
warning: redirecting to https://github.com/DivyaKumar-19/GitDemo.git/
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 228 bytes | 228.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To http://github.com/DivyaKumar-19/GitDemo.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.
```

Git Repository



2. Git-HOL

```
MINGW64/c:/Users/Divya/GitIgnoreDemo
$ cd GitIgnoreDemo

Divya@LAP19 MINGW64 ~/GitIgnoreDemo
$ git init
Initialized empty Git repository in C:/Users/Divya/GitIgnoreDemo/.git/

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ echo "This is a log file">debug.log

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ mkdir log

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ echo "Another log file">log/error.log

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    debug.log
    log/

nothing added to commit but untracked files present (use "git add" to track)

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ echo ".log">.gitignore

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ echo "log/">.gitignore

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ git commit -m "Add gitignore to ignore log files and log folder"
[master (root-commit) e31616] Add gitignore to ignore log files and log folder
1 file changed, 2 insertions(+)
create mode 100644 .gitignore

Divya@LAP19 MINGW64 ~/GitIgnoreDemo (master)
$ git status
On branch master

nothing to commit, working tree clean
```

3. Git-HOL

```
MINGW64/c/Users/Divya/GitBranchLab
Divya@LAP19 MINGW64 ~
$ mkdir GitBranchLab
Divya@LAP19 MINGW64 ~
$ cd GitBranchLab
Divya@LAP19 MINGW64 ~/GitBranchLab
$ git init
Initialized empty Git repository in C:/Users/Divya/GitBranchLab/.git/
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ echo "This is the master file" > masterfile.txt
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git add masterfile.txt
warning: in the working copy of 'masterfile.txt', LF will be replaced by CRLF the next time Git touches it
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git commit -m "Initial commit on master"
(master root-commit) 1928c1e Initial commit on master
1 file changed, 1 insertion(+)
create mode 100644 masterfile.txt
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git branch GitNewBranch
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git branch
  GitNewBranch
* master
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git checkout GitNewBranch
Switched to branch 'GitNewBranch'
Divya@LAP19 MINGW64 ~/GitBranchLab (GitNewBranch)
$ echo "This is a file in GitNewBranch" > branchfile.txt
Divya@LAP19 MINGW64 ~/GitBranchLab (GitNewBranch)
$ git add branchfile.txt
warning: in the working copy of 'branchfile.txt', LF will be replaced by CRLF the next time Git touches it
Divya@LAP19 MINGW64 ~/GitBranchLab (GitNewBranch)
$ git commit -m "Add branchfile.txt in GitNewBranch"
[GitNewBranch e17afd1] Add branchfile.txt in GitNewBranch
1 file changed, 1 insertion(+)
create mode 100644 branchfile.txt
```

```
MINGW64/c/Users/Divya/GitBranchLab
Divya@LAP19 MINGW64 ~/GitBranchLab (GitNewBranch)
$ git add branchfile.txt
warning: in the working copy of 'branchfile.txt', LF will be replaced by CRLF the next time Git touches it
Divya@LAP19 MINGW64 ~/GitBranchLab (GitNewBranch)
$ git commit -m "Add branchfile.txt in GitNewBranch"
[GitNewBranch e17afd1] Add branchfile.txt in GitNewBranch
1 file changed, 1 insertion(+)
create mode 100644 branchfile.txt
Divya@LAP19 MINGW64 ~/GitBranchLab (GitNewBranch)
$ git checkout master
Switched to branch 'master'
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git diff master GitNewBranch
diff --git a/branchfile.txt b/branchfile.txt
new file mode 100644
index 0000000..70ab14d
--- /dev/null
+++ b/branchfile.txt
@@ -0,0 +1 @@
+This is a file in GitNewBranch
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git merge GitNewBranch
Updating 1928c1e..e17afd1
Fast-forward
 branchfile.txt | 1 +
 1 file changed, 1 insertion(+)
create mode 100644 branchfile.txt
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git log --oneline --graph --decorate
fatal: unrecognized argument: --oneline
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git log --oneline --graph --decorate
* e17afd1 (HEAD -> master, GitNewBranch) Add branchfile.txt in GitNewBranch
* 1928c1e Initial commit on master
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$ git branch -d GitNewBranch
Deleted branch GitNewBranch (was e17afd1).
Divya@LAP19 MINGW64 ~/GitBranchLab (master)
$
```

4. Git-HOL

```
MINGW64/c/Users/Divya/GitConflictLab
Divya@LAP19 MINGW64
$ mkdir GitConflictLab
Divya@LAP19 MINGW64 ~
$ cd GitConflictLab
Divya@LAP19 MINGW64 ~/GitConflictLab
$ git init
Initialized empty Git repository in C:/Users/Divya/GitConflictLab/.git/
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ echo "Initial file"> start.txt
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git add start.txt
warning: in the working copy of 'start.txt', LF will be replaced by CRLF the next time Git tou
ches it
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git commit -m "Initial commit"
[master (root-commit) f2e4ff6] Initial commit
1 file changed, 1 insertion(+)
create mode 100644 start.txt
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git status
On branch master
nothing to commit, working tree clean
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git branch GitWork
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git checkout GitWork
Switched to branch 'GitWork'
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ echo "<message> Hello from GitWork</message>"> hello.xml
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git tou
ches it
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ git commit -m "Add hello.xml in GitWork branch"
> git commit -m "Add hello.xml in GitWork branch"
> git commit -m "Add hello.xml in GitWork branch"
git commit -m "Add hello.xml in GitWork branch"
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ git checkout master
Switched to branch 'master'
```

```
MINGW64/c/Users/Divya/GitConflictLab
$ echo "<message> Hello from GitWork</message>"> hello.xml
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git tou
ches it
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ git commit -m "Add hello.xml in GitWork branch"
> git commit -m "Add hello.xml in GitWork branch"
> git commit -m "Add hello.xml in GitWork branch"
git commit -m "Add hello.xml in GitWork branch"
error: pathspec 'in' did not match any file(s) known to git
error: pathspec 'GitWork' did not match any file(s) known to git
error: pathspec 'branch' did not match any file(s) known to git
error: pathspec 'branch' did not match any file(s) known to git
git commit -m Add' did not match any file(s) known to git
error: pathspec 'in' did not match any file(s) known to git
error: pathspec 'GitWork' did not match any file(s) known to git
error: pathspec 'branch' did not match any file(s) known to git
[GitWork c446a59] Add hello.xml in GitWork branch
1 file changed, 1 insertion(+)
create mode 100644 hello.xml
Divya@LAP19 MINGW64 ~/GitConflictLab (GitWork)
$ git checkout master
Switched to branch 'master'
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ echo "<message> Hello from master branch</message>"> hello.xml
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git add hello.xml
warning: in the working copy of 'hello.xml', LF will be replaced by CRLF the next time Git tou
ches it
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git commit -m "Add hello.xml in master branch"
[master bbf7819] Add hello.xml in master branch
1 file changed, 1 insertion(+)
create mode 100644 hello.xml
Divya@LAP19 MINGW64 ~/GitConflictLab (master)
$ git merge GitWork
Auto-merging hello.xml
CONFLICT (add/add): Merge conflict in hello.xml
Automatic merge failed; fix conflicts and then commit the result.
Divya@LAP19 MINGW64 ~/GitConflictLab (master|MERGING)
$
```

5. Git-HOL

```
MINGW64/c/Users/Divya/GitCleanupLab
Divya@LAP19 MINGW64 ~
$ mkdir GitCleanupLab
Divya@LAP19 MINGW64 ~
$ cd GitCleanupLab
Divya@LAP19 MINGW64 ~/GitCleanupLab
$ git init
Initialized empty Git repository in C:/Users/Divya/GitCleanupLab/.git/
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ echo "Initial file"> start.txt
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git add start.txt
warning: in the working copy of 'start.txt', LF will be replaced by CRLF the next time Git tou
ches it
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git commit -m "Initial commit"
[master (root-commit) 8344241] Initial commit
1 file changed, 1 insertion(+)
create mode 100644 start.txt
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git remote add origin https://github.com/Divyakumar-19/GitCleanupLab.git
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 227 bytes | 227.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Divyakumar-19/GitCleanupLab.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git branch
* master
```

```
MINGW64/c/Users/Divya/GitCleanupLab
$ echo "Initial file"> start.txt
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git add start.txt
warning: in the working copy of 'start.txt', LF will be replaced by CRLF the next time Git tou
ches it
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git commit -m "Initial commit"
[master (root-commit) 8344241] Initial commit
1 file changed, 1 insertion(+)
create mode 100644 start.txt
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git remote add origin https://github.com/Divyakumar-19/GitCleanupLab.git
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 227 bytes | 227.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Divyakumar-19/GitCleanupLab.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git branch
* master
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git pull origin master
From https://github.com/Divyakumar-19/GitCleanupLab
 * branch master -> FETCH_HEAD
Already up to date.
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$ git push origin master
Everything up-to-date
Divya@LAP19 MINGW64 ~/GitCleanupLab (master)
$
```