1. **How to run the pytest in command line?**

Give the path and the test name.

k for method name execution, s for logs in output, v stands for metadata.

1. **How to group the test like smoke or unit?**

Mark the test with tag like @pytest.mark.smoke and run with -m.

1. **How to skip a test?**

@pytest.mark.skip

1. **How to run a test and not report?**

@pytest.mark.xfail

1. **Git commands**

**To Push the code from the system to the github – Go to cmd**

Tell git who you are - git config --global user.name “Himanshu”

git config –global user.email “[hsethi@bgsu.edu](mailto:hsethi@bgsu.edu)”

Go to the folder like C:\\Users/Himanshu\\AppiumFramework and the use the following code

Create a new local repository - git init

Add files – git add \* / git add .

Commit – git commit -m “Message”

Connect to repository – git remote add origin File path from github

Push – git push origin master

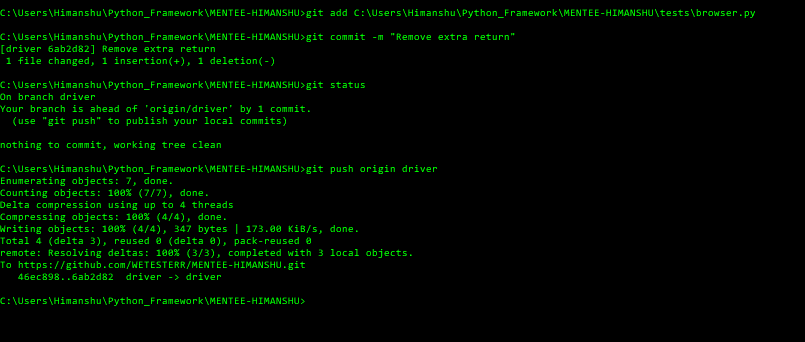
**To get the codes from github to the system for the first time**

git clone path from the github

**To get the codes from github to the system after that**

git pull path from the github

**After making the changes to the code and push it back to github**



Add files – git add filepath

Commit – git commit -m “Message”

Connect to repository – git remote add origin File path from github

Push – git push origin master

1. Create the branch name: Go to the project path….

git branch branch name

Then you have to push this branch to the github server either from pycharm or cli

1. Before creating new branch---- git checkout master

git pull

1. To check the number of branches -- git branch
2. **Always switch to the branch or master using git checkout branch name or master.**
3. We can also create a branch with git checkout -b branch name
4. We can get the status where we are ----- git status
5. Then you have to push this branch to the github server
6. To delete a branch – git branch -d branch name

**Git commands in pycharm**

1. Control +k to commit
2. Control + shift + k to push

**Allure Commands**

1. To enable Allure listener to collect results during the test execution simply add --alluredir option and provide path to the folder where results should be stored. E.g.:

--alluredir=../my\_allure\_results

So the full command with the other input variables would be: --browser\_name chrome –email [hs002@gmail.com](mailto:hs002@gmail.com) –password –abc@123 –alluredir=../my\_allure\_results

1. To see the actual report after your tests have finished, you need to use Allure commandline utility to generate report from the results.

allurepath serve ../my\_allure\_results

Here the allurepath = C:\Users\Himanshu\Desktop\Automation\Python\_Selenium\allure-2.13.5\bin\allure.bat