Git bash tutorials

Tutorial 2

1. Git config –global user.name ‘Henry Salim’
2. Git config –global user.email ‘henrysalim31@gmail.com
3. Git init .
4. Touch clish.txt
5. Add some lines into the clish.txt file
6. Git status
7. Git add clish.txt
8. Git commit -m “added a new file”
9. Git log
10. add a new line to clish.txt
11. Git add clish.txt file
12. Git status should say modified a new line in the status
13. Git commit -m ‘added the file again’
14. Git log
15. And that is it for this tutorial I hope you learnt a lot in this video I look forward to teaching you more Clinton

Tutorial 3-Branches

1. To create a branch in git use git branch Develop
2. To check the branches that you have git branch
3. To switch to another branch git checkout develop
4. Here I learned how to use the -am to add and commit the changes at the same time.
5. To create a branch and check out to it at the same time use git checkout -b feature/new-feature
6. And that’s it for branches next we’ll look at out we can merge branches.

Tutorial 4 -Merging branches

1. To merge branches you have to go back to the lower level branch and type
2. Git merge feature/newfeature
3. You can look at what you have merged with git show and the merge id
4. This will create a merge commit so you don’t have to add and commit the process
5. Ooh one more thing to get the id of the merge you have to type the usual git log to see it

Tutorial 5- Git flow to see how it all connects.

1. To initialize git flow you have to have git initialized in the directory you are working on ie. Git init . .
2. Then you have to install the git-flow by brew install git-flow
3. Then if you want to use the git flow just do git-flow init -d
4. That will use the default settings of the git flow
5. Then you will want to delete the new-branch that we created earlier by – git-flow feature finish new-feature
6. Would you look at that in the recent productions of the git the git-flow is already integrated into the program. How nice as my supervisor would say.
7. Another thing to initialize git-flow you have to type git-flow init leave alone the previous readings
8. And that’s all you have to know for now see you in the next tutorial where we’ll look at how to work with git and publish in github.

Tutorial 6-creating the ssh key new feature and the pushing to the git repositories

1. First if you want to create a new-feature branch you just type feature start new-feature
2. The you can checkout to that branch
3. Then you will need to create an ssh key to put into your github so that github knows that you’re the one pushing stuff to your git account repository
4. To create an ssh-key type ssh-keygen -t rsa -C ‘your email’
5. To save changes as you already know you will use git commit -am ‘Blablabla’
6. Then you will push then to your git account
7. But before you push them to your git account you will have to add the git ssh link to your local git account to do this just type git remote add origin ‘then your git ssh’
8. Then you can check all your remote repositories by typing git remote -v
9. To push a work commit to your repo just type git push origin branch
10. And that basically it and if you want someone else to be able to push to that repo you will need to share your git public key with them
11. Note: don’t forget to google what the the git add -A does or I think you should be knowing by

now

1. You forgot about git flow feature publish a-new-feature

Tutorial 7- in this tutorial you will learn about the pull request and release branches

1. Here what I learned is that you can create a new pull request from the git hub page and do some comments. And if you’re pulling from the develop branch you can also select the freature branceh to work on since it comes from the develop branch
2. And if you finish the the new-feature in feature finish new-feature you will be prompted to merge it to the develop branch which is a good thing since we want that to happen somehow
3. Then you have to update the same to github. Just do a push request to github and it will automatically detect that you merged to branch develop
4. And that’s it for this tutorial. But I think you have a lot to learn about pull request so that you can work on another persons git hub repo.