cd C:

**/c** **Command**: git clone [git@github.com:YottaMark/QA.git](mailto:git@github.com:YottaMark/QA.git)

cd QA

Open bash shell Run

git add –u (declare your changes to local repo)

**Commit all new files to local version control.**

git commit –m "Just deleted Files" (commit locally)

**Push files to GitHub**

git push (send out to GitHub)

**Remove file before you commit. Run Command:**

git rm hello.txt

**So what if I need to delete pushed files:**

git add . –u (announce changes to local)  
git commit –m "Just deleted Files" (commit locally)   
git push (send out to GitHub)

**How to create new branch locally and push it to GitHub?**

**(Note –b use this option to create new branch only) Open bush shell**  
git checkout -b <newbranchname>  
git push origin newbranchname (Just created new branch in GitHub)   
git branch --set-upstream newbranchname origin/newbranchname This command will link your local branch with GitHub

* **How do I get changes from GitHub master?**

Move to master and issue **Command**:

git pull

**You have done your work on your local branch, so next you need to push your changes to GitHub master.**

Make sure you are in your feature add, commit and push your chances   
Now you should synchronize with master.   
Move to master   
 git pull  
Switch to feature branch (git checkout feature branch)   
 git merge master   
push to feature branch (remote)  
 git push   
Move to master   
 git merge feature  
Move to master   
 git push

**Before you start working on your code , you need to get all new changes from GitHub master merge them to your local feature branch. Get History Command:**

git log -n 3   
Move to master   
 git pull  
Now you've just got new changes in your local "master".

Now you have to merge new changes with your feature (working branch).  
Move to feature branch.

git merge master

Now your code up-to-date and you ready to work on your new code.

**How to move between branches?**

git checkout master (move to master but no merge)

**Delete Branch from Remote repo:**

git push origin --delete <branch name>

**Tags and how to tag it:**

$ git tag v1.3.1

$ git tag  -a "v1.3­.2" -m "Vers­ion v1.3.­2"

$ git push --tag­s

To https:/­/git@pshop­.com/petsh­op\_online/­petshop.gi­t  
 \* [new tag]         v1.3.1 -> v1.3.1  
 \* [new tag]         v1.3.2 -> v1.3.2

Success!

To get back to old tag

$  git checkout 1.3.1 (old tag)