

## **How to work with git in windows [using git bash]**

Commands:

Dir => to find all subfolders under current folder

### **1. How to navigate between directories?**

Cd /d/te4b/HsseMobile

**or**

Cd d:

Cd te4b

Cd HsseMobile

### **2. How to create a subfolder?**

Mkdir <Foldername>

**Or**

Mkdir /d/te4b/HsseMobile

### **3. How to check info of a command?**

Git help <command\_name>

Ex:

Git help branch

### **4. How to see commonly used git commands?**

Git help

### **5. Work with frequently used git commands**

Useful steps in team work

**Step1: Create folder in your computer let's say in D drive </d/te4b/HoseMobile>**

\$ mkdir /d/te4b/HoseMobile

**Stpe2: navigate to the drive**

Git /d/te4b/HoseMobile

**Step3: Clone your repository in the directory**

git clone <git uri>

Enter username and password

#### **Step 4: Check in which branch you are ?**

`$ git branch --list`

**Or**

`$ git branch`

**Or**

`$ git branch --all`

Existing branches are listed; the current branch will be highlighted with an asterisk

#### **Step 5: Navigate to a new branch from current branch**

`git checkout <branch name>`

Ex: `git checkout release/rc1`

**Note:** before you checkout make sure you don't have any uncommitted changes in your branch

#### **Step6: How to commit new file or modify existing file in the repository**

1. `git status`
2. `git add . ==>` to stage all unstaged files

**OR**

`git add <file path1><space><file path1> ==>` to stage specific unstaged file

3. `git commit -m <"message">`
4. `git push`

**OR**

`git push origin <branch> ==>` push to a particular branch changes done in current branch

#### **Step 7: How to merge develop[current] branch with another branch[feature/hamburgermenu] ?**

`git checkout develop`

`git pull`

`git merge feature/hamburgermenu [Resolve conflicts if any ]`

`git push`

#### **Q.How to delete a local branch**

```
$ git branch -d branch_name
```

### **Step 8: How to use git stash command?**

git checkout develop

git pull

Let's say you are in develop branch

You have done some changes in current branches, after a while you remember that you should not have done changes in this branch, Then how can you put your changes to new branch without losing your changes?

git stash

git checkout -b <branch name>

git stash pop

Git status

git add .

Git commit -m "message"

Git push

### **Q How to clear modified/Added/Removed files from locally using git**

\$ git reset --hard

Reference:

<http://stackoverflow.com/questions/7770364/git-undo-local-changes-git-add-git-rm>

Q How to create new branch from current branch

<http://stackoverflow.com/questions/4470523/git-create-a-branch-from-another-branch>

**you have modified 4 files, let's say f1,f2,f3,f4 But when you you want to push the code to remote you don't want to commit f1 and f2 files.**

**How to revert unstaged files?**

<http://stackoverflow.com/questions/14075581/git-undo-all-uncommitted-changes>

\$ git status //check all files are unstaged

// //How to revert uncommitted changes only to particular file or directory:

\$ git checkout -- <f1 path>

```
$ git checkout -- <f2 path>
```

```
$ git commit -m "Modified files for Report Summary"
```

```
$ git pull
```

```
$ git push
```

### **how to revert staged files ?**

```
$ git status
```

```
$ git add . //now all files are staged
```

```
$ git reset <f1 path>
```

```
$ git reset <f2 path>
```

```
git status
```

```
$ git commit -m "Modified files for Report Summary"
```

```
$ git pull
```

```
$ git push
```

### **How to commit changes done in current branch to new branch without losing your changes?**

```
git status
```

```
git stash
```

```
git checkout develop
```

```
git pull
```

```
git checkout -b <branch name>
```

```
git stash pop
```

```
Git status
```

```
git add .
```

```
Git commit -m "message"
```

```
Git push
```

### **How to come back from merge page to back**

Enter following keys in order

<ESC>

:

W

Q

<Enter>

### **How to merge and resolve merge conflicts**

```
git status
```

```
git checkout develop
```

```
git pull
git merge <feature/rabcintegration>
--You will get following message
Auto-merging Ina.Services.HSSE.Entities/Ina.Services.HSSE.Entities.csproj
Auto-merging Ina.Services.HSSE.Api/Web.config
CONFLICT (content): Merge conflict in Ina.Services.HSSE.Api/Web.config
Automatic merge failed; fix conflicts and then commit the result.
-- From the above message Ina.Services.HSSE.Api/Web.config file got conflict
-- So go to that file and remove or add whatever code you want
--Resolve conflicts in all files.
git status
--Now it will show all files you modified when you resolving conflicts
--stage files by giving the following command
git add <filename>
or
git add .
git commit -m "message while resolving conflict"
git push
git status

--You merged successfully
```

### **Q How to remove a modified and unstaged file/class from list of unstaged files**

//Check what changes you have done with the following command

```
$ git status
```

On branch feature/dynamicuimetadata

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

```
modified: HSSEMobile/HSSEMobile.Droid/HSSEMobile.Droid.csproj
modified: HSSEMobile/HSSEMobile.Droid/Resources/Resource.Designer.cs
modified: HSSEMobile/HSSEMobile/App.cs
modified: HSSEMobile/HSSEMobile/HSSEMobile.csproj
modified: HSSEMobile/HSSEMobile/Views/AboutPage.cs
modified: HSSEMobile/HSSEMobile/Views/FormSelectView.cs
modified: HSSEMobile/HSSEMobile/Views/LoginPage.cs
modified: HSSEMobile/HSSEMobile/Views/ProjectSelectionPage.cs
modified: HSSEMobile/HSSEMobile/Views/SectionSelectPage.cs
```

[Let's say .Droid.csproj and Resource.Designer.cs files get modified but you don't want to commit]

//compare with earlier code

```
$ git diff *.Droid.csproj
$ git checkout *.Droid.csproj
$ git diff *.Designer.cs
$ git checkout *.Designer.cs
$ git status
$ git add .
$ git status
$ git commit -m "Added Folders,classes,Methods required for making Dynamic UI text"
$ git status
$ git pull
$ git push --set-upstream origin feature/dynamicuimetadata
$ git push
```

### **Q How to do small commits**

```
git add JobManager/Controllers/APIs/ProfileApiController.cs
```

```
git commit -m "your message"
```

```
git push
```

OR

```
git commit file1 file2 file5 -m "commit message"
```

### **Q.How to Check Repository URL that you are working in Git bash?**

```
$ git remote show origin
```

Q How to push commit and not pushed commits to different branch?

```
$ git status
```

```
$ git add .
```

```
$ git commit -m "1. updated sqlite related packages, and added one new package microsoft.bcl
2. added logic for projectlist autocompletion"
```

```
$ git reflog
```

```
$ git log --graph --decorate --abbrev-commit --all
```

```
$ git reset --soft HEAD^
```

```
$ git checkout -b feautre/autocompleteproject
```

```
$ git status
```

```
$ git add .
```

```
$ git commit -m " 1. updated sqlite related packages, and added one new package microsoft.bcl  
2. added logic for projectlist autocompletion"
```

```
$ git push
```

### **How to tag**

```
$ git tag v1.0.0-RC1
```

### **How to delete tag**

```
$ git tag -d v1.0.0-RC1
```

### **How to reset latest commit once it is staged**

```
$ git reset --soft HEAD^
```

### **How to see commit info**

```
$ git log --graph --decorate --abbrev-commit --all --name-status
```

### **HOW to reset configuration done in git in MAC device**

```
git config --global --unset-all user.name
```

```
git config --global --unset-all user.email
```

### **How to config user in git mac:**

```
git config --global user.name "karunakar"
```

```
git config --global user.email karunakar.b@techjini.com
```

### **How to do soft reset of commit you have done**

```
git reset --soft HEAD^
```

## How to head to 19b5d5f by ignoring 402cf4b commit or ignore commits

1. Take one branch from develop **origin/feature/singlefileupload**
2. Do some commits in this branch
3. Some body committed to develop branch and you want to ignore commits done by
4. someone in develop and merge your branch to develop
- 5.

```
$ git log --graph --full-history --all --pretty=format:"%h%x09%d%x20%s"
* 0c59f50      (HEAD -> origin/feature/singlefileupload
* 232bdeed    Modified to resolve staging issues origin/feature/singlefileupload
* 232bdec     Added config transformation for qa and staging and modified all corresponding
end points in we.config files origin/feature/singlefileupload
| * 402cf4b    (tag: v1.0.0-RC2) Added Staging Configuraiton,origin/develop,
|/
* 19b5d5f     modified files to prefix schema to all sps
* 460061a     modified files to handle all failure cases in sync up process
```

```
$ Git checkout develop
$ git reset --hard HEAD^
$ git status
$ git merge feature/Singlefileupload
```

## How to check Git logs

```
gitk
git show ls-remote
```

```
git show ls -remote
git ls-remote
```

```
git log --graph
```







