# GITHub Commands: Getting to know GitHub

After you install the required software, run through these tasks to become familiar with the basics of GitHub, our source-code reposititory. You'll need to understand and feel comfortable performing these tasks before you can start editing and contributing to WebLabUX source code.

## Create a new repo and save its first file

|  |  |
| --- | --- |
| Action | Command |
| Open git shell | (depends on OS) win: cmd or open Git Shell mac: |
| Go to root folder | cd <your git root> (This is automatic if you open the Git Shell) |
| Create new project (repo) | git init MyFirstRepo |
| Go to project folder | cd MyFirstRepo |
| Create a new text file | (depends on OS) win: notepad newfile1.txt mac: |
| Enter text and save file | "hello world" |
| Add changes to commit | git add newfile1.txt |
| commit changes | git commit -m "created hello world file" |
| view file | more newfile1.txt |

### Create a branch and edit your file in it

|  |  |
| --- | --- |
| Action | Command |
| Create a branch called "Work" | git branch Work |
| Use that branch | git checkout Work |
| Edit the file in that branch | (depends on OS) Win: notepad newfile1.txt |
| Change text and save changes | (change "world" to "venus") |
| Add the file to the commit list | git add newfile1.txt |
| Commit the changes | git commit -m "changed world to venus" |
| View the file | more newfile1.txt (see "Hello venus") |
| Go back to the master branch | git checkout master |
| View the file | more < newfile1.txt (see "Hello world") |

## Edit the file in the Master branch

|  |  |
| --- | --- |
| Action | Command |
| Go to the Master branch | git checkout Master |
| Edit the file | (depends on OS) Win: notepad newfile1.txt |
| Change the text and save | (change "world" to "mars") |
| Add the file to the commit | git add newfile1.txt |
| Commit the changes | git commit -m "changed world to mars" |
| view the file in Master | more newfile1.txt (see "Hello mars") |
| Switch to Work branch | git checkout Work |
| view the file in Work | more < newfile1.txt (see "Hello venus") |
| Switch to Master branch | git checkout Master |

## Merge the two branches

|  |  |
| --- | --- |
| Action | Command |
| Start in the Master branch | git checkout Master |
| Start merge process to merge the Work branch with the Master branch | git merge work |
| See file with merge conflicts | more newfile1.txt |
| Open merge tool to resolve conflict | git mergetool |
| Resolve conflict in tool by selecting "Hello Venus" |  |
| Save changes in merge tool |  |
| Commit the merge results | git commit -m "merged Work branch into Master" |
| Remove any lingering files from the merge | del \*.orig |

## Edit the file in Work and commit several changes

|  |  |
| --- | --- |
| Action | Command |
| Start from Master branch | git checkout Master |
| View file in Master branch | more newfile1.txt |
| Go to Work branch | git checkout Work |
| View file in Work branch | more newfile1.txt |
| Edit file | (depends on your OS) Win: notepad newfile1.txt Mac: |
| Change world and save changes | (change "venus" to "io") |
| Commit the changes | git commit -a -m "moved to io" |
| Edit the file again | (depends on your OS) Win: notepad newfile1.txt Mac: |
| Change the world and save changes | (change "io" to "Europa") |
| Commit the changes | git commit -a -m "moved to Europa" |
| Edit the file one more time | (depends on your OS) Win: notepad newfile1.txt Mac: |
| Change the world and save changes | (change "Europa" to "Callisto") |
| Commit the changes | git commit -a -m "moved to Callisto" |

## Collapse the commits in the Work branch

|  |  |
| --- | --- |
| Action | Command |
| Squash the last three commits into one | git rebase -i HEAD~3 |
| Review the commits you see in the editable window that opens. It should have a line for each commit that says "pick," <commit number>, and your commit comment. | Leave the top line alone.  Change "pick" to "squash" in the next two lines.  Save the file. |
| See that git is still working on your command and opens another file to edit | Add a comment that describes the net result of the commits you're merging. Lines that start with a "#" are ignored.  The lines that don't start with a "#" will be used as your comment for this commit/squash.  If you don't have a comment line, the action will stop and the commits won't be merged. |
| Wait for the command line to return |  |

## Merge the Work branch changes into Master

|  |  |
| --- | --- |
| Action | Command |
| Go to Master branch | git checkout Master |
| Merge the work branch | git merge Work |