**Git Commands and Parameters (In English) changed**

1. git clone - Makes a Git repository copy from a remote source.
2. git branch - List the branches that you have locally.
3. git checkout [branch name] - Switches between branches.  Note: you can not switch branches if it will result in the loss of data.
4. git checkout [file name] - Revert any changes made to the file in the working copy by overwriting the file with the version in the local repository.
5. git pull --no-ff origin [branch name]  - Does a git fetch to get the changes from the remote repository into the local repository followed by a git merge to move the files from the local repository into the working copy. This will bring your current repository and working copy up to date with the remote repository.  Note: This command is discouraged as it does two operations.
6. git fetch - retrieves all the changes from the remote repository into the local repository.
7. git merge - merges changes from the local repository into the working copy.
8. git checkout -b [project name] - Create a new branch or if  the branch already exist remotely it will map it (make it visible for use).
9. git add --all - Stage the files in your working copy so the 'git commit' command can be used to add them to the repository.
10. git status - Shows files that are untracked (only in your working copy), modified (tracked but not yet committed), and staged (added and ready to be committed).
11. git commit -m "some description" - Commit the files that have been staged into the repository.
12. git push origin [branch name] - Push the changes that have been committed into the local repository up to the remote repository.
13. git pull --no-ff origin [branch name] - Retrieve the most current files from the remote repository and merges it with your local repository.
14. "." (without the quotes).  This will perform the operation on the current directory and all sub directories.  eg. if "git checkout ." is executed from the root directory all unstaged changes will be lost.
15. To set your branch to exactly match the remote branch perform the following 2 git commands:
    1. git fetch origin
    2. git reset --hard origin/[branch name]
16. git branch -d [local branch] - Deletes a local branch.
17. git push origin :[remote branch] - Deletes a remote branch.
18. git rm [file name] - Removes a file and stages ALL changes made so far (files in work and deleted file) in working.  If you have multiple files,  execute this for each file.