# Create New Remote Branch

To set up a new remote branch (from your local), make sure you are on the desired branch then run the following command.

git branch -u <remote>/<branch>

# Create New Local Branch (and switch to it)

Allows you to create a new local branch (of your current state) and immediately switch to using the new one.

git checkout -b <branchname>

# Rename Local Branch

To rename a local branch, first check it out and then run the following.

git branch -m <new-name>

# Rename Remote Branch

To rename a remote branch, first rename the local branch as desired (described above). Then delete the old-name remote branch and push the new-name local branch.

git push origin :<old-name> <new-name>

Finally, to reset the upstream branch for the new-name local branch, switch to the branch and then run the following command.

git push origin -u <new-name>

# Delete Local Branch

To remove a local branch (say after merging it into another branch), use the following command:

git branch -d <branch-name>

If you need to force remove the branch, then use an uppercase D in the same command.

git branch -D <branch-name>

# Delete Remote Branch

*Note: Only perform this action once it is safe to do so, such as the remote branch has been successfully merged into the master (production) branch.*

git push origin --delete <remote>

# Remove Stale Remote References

When a remote branch has been deleted, your local reference to that branch does not get removed automatically. So after a while, you may have several branch references for remote branches that no longer exist. To clean up those outdated (stale) references:

git fetch -p

# Amend Previous Commit with New Content

Allows you to include additional changes with your latest commit.

git commit --amend

# Amend Previous Commit with Current Date and Time

Allows you to update the timestamp associated with your most recent commit.

git commit --amend --date="$(date)"

# Reorder Commits (and other actions)

Allows you to go and edit the order of your git commits, combined individual commits, and modify the commit messages before pushing to the repository. If the branch is not specified, it will attempt to rebase against the default (master) branch.

git rebase -i <branch>

# Rewrite History (Remote Commits)

So you fancy yourself a Time Lord and have decided to change the course of history itself. Well, you can alter the git commits that are already pushed into the repository by rebasing against a past version of the the branch. Then force-push the altered commits back into the repository to change the tapestry of history itself! Be careful though as this could potentially have unforeseen consequences for commits after the one(s) you altered. *Use this power responsibly!*

git rebase -i HEAD~<number backwards> or git rebase -i <commit hash>

git push -f

# Push Commit(s) up to Specified Point

This will push all your local commits to the git repository, up to and including the specified commit.

git push <remote> <commit hash>:<branch>

For example:

git push origin db33a677be07542cfae8bf58757460572964c35e:master

This feature is best used in conjunction with the *git rebase -i* to essentially cherry-pick which commits (and their order) you want to push up to the repo.

# Move Branch with Head to a Point

This will allow you to move up a branch tag with the head tag at the same time to a specific point in the tree. This happens most often when we are done testing something in staging and want to move master along.

git branch -f <branch> <commit hash>

# Setting the default editor for Git

Allows you to set which text editor you want to use by default with git. Ryan is a big fan of *nano* while Adam prefers *vim*.

git config --global core.editor "vim"

# Additional Resources

* [Git Simple Interactive Tutorial](https://try.github.io/levels/1/challenges/1)
* [Git Branching Tutorial](https://www.atlassian.com/git/tutorials/using-branches/)
* [Git Branching Interactive Tutorials](http://learngitbranching.js.org/)
* [Git Command Reference](http://gitref.org/)
* [Git Push Up to Commit](https://coderwall.com/p/hexinq/git-push-up-to-a-certain-commit)
* [Git Simple Guide](http://rogerdudler.github.io/git-guide/)