Git: merging a development branch into master

> This is a work-in-progress pattern for merging a development branch into master - we should apply this pattern consistently for our magnolia projects on Git. This is necessary to help keeping the Git history as clean as possible, i.e. without empty merge commits.

Prerequisites

- local master is up-to-date

Steps

1. First create an *integration* branch out of your development branch
   ```
   git checkout -b myfeature-int
   ```
   a. This guarantees that you won't spoil your development branch
   b. This is especially valid for dev branches that are also remote, we should never rewrite the history of a shared branch

2. Then **rebase** that new branch onto master
   ```
   git rebase -i master myfeature-int
   ```
   a. Ideally you do an interactive rebase with the -i flag, so that you can squash, reorder, or drop commits
   b. Having 3 commits for subsequently renaming a class is an ideal candidate for squashing
   c. Don't hesitate to re-do it several times over. It's easier to, for example, first reorder some commits, make sure they can be re-applied in that order, then do a second rebase to squash'em.
3. Publish that integration branch, so this can be used for reviews
   ```
   git push -u origin myfeature-int
   ```
4. Now all your commits are stacked on top of the latest master, therefore merging to master should be a fast-forward. We want to enforce this.
   ```
   git checkout master
   git merge myfeature-int --ff-only
   ```
   a. This will reject non fast-forward merges

5. Finally if you have to pull again from master before pushing, don't forget to do a *pull rebase*.
   ```
   git pull --rebase
   ```

6. Delete your integration branch
   ```
   git branch -d myfeature-int
   ```
   If you pushed it, then delete the remote branch too: `git push origin --delete myfeature-int`

Result

- Your branch commits are flattened into master's history
- You can keep working on your dev branch
  - remote tracking is not spoiled by the rebase
  - then reapply the same pattern for the next merge to master

Success!
CLAIRE'S WORLD