**Git NOTE**

**GIT BRANCHING**

generalizations

Let’s take a moment to review the main concepts and commands from the lesson before moving on.

* Git *branching* allows users to experiment with different versions of a project by checking out separate *branches* to work on.

The following commands are useful in the Git branch workflow.

* git branch: Lists all a Git project’s branches.
* git branch branch\_name: Creates a new branch.
* git checkout branch\_name: Used to switch from one branch to another.
* git merge branch\_name: Used to join file changes from one branch to another.
* git branch -d branch\_name: Deletes the branch specified. This is used at the end of the merging task when the branch already served its purpose.

Practically, we:

1. Create a new branch and often name it after our purpose, e.g. big-head that enlarge the size of head.
2. Start modify our file and add it to staging area (git add <filename>)
3. Make a commit for that
4. Merge the branch --- and switch back to the master branch (after this usually followed by deleting the created branch when the purpose is served)

**Teamwork**

In order to get your own replica of repository, you’ll need to *clone* it with:

git clone remote\_location clone\_name

After this, change directory (cd …) to the cloned repo on the local device by

And a list of a Git project’s remotes can be seen with the command:

git remote -v

It’s easy to see if changes were already made to the remote and bring the changes down to your local copy is with:

git fetch

A flow of team-up works on Git follows the process:

* 1. Fetch and merge from the remote repository to our local device.
  2. Create a new branch to add some features.
  3. Commit the features.
  4. Do step 1 again, that fetch and merge from the remote repository, in case to avoid some new cases that happened during our working time.
  5. Push our branch up to review (share our works with the colleagues, whilst avoiding merge conflicts).
* git clone: Creates a local copy of a remote.
* git remote -v: Lists a Git project’s remotes.
* git fetch: Fetches work from the remote into the local copy.
* git merge origin/master: Merges origin/master into your local branch. The version/changes/features will from colleagues will be updated to the local copy.
* git push origin <branch\_name>: Pushes a local branch to the origin remote.