Create a branch with git branch [branch name]

To switch to this branch you use git checkout [branch name]

To create a branch AND switch to it use git checkout –b [branch name]

Naming a branch is usually with the work you are carrying out

Switching branches means Git replaces your working directory with the snapshot of the latest commit on that branch .

This means you don't have to have multiple directories for multiple branches.

Once you have finished working on your branch you will eventually want to incorporate it into your main branch.

You merge branches together with git merge.

However, this doesn’t mean you won’t still get merge conflicts.

For example, if the same line has been changed in two different branches – lets call them branch 1 and branch 2, if you try and merge branch 1 with branch 2 you will get a merge conflict. You will see the standard merge conflict markers in your text editor and you have to amend these manually.

Once manually amended it may be worth running git diff as this shows you both sides of the conflict and how you've resolved it

Once happy with your branch you would add and commit in the normal way, but if you try to push it to your remote version it won’t work as it doesn’t exist yet.

The way to get around this is using the command git push –u origin [branch name]

Once finished with a branch it is worth deleting it and creating a new one.

If we want to delete a you can run the command git branch -d (branch) to remove it.