Branching: <https://www.atlassian.com/git/tutorials/using-branches>

In Git, a branch is a new/separate version of the main repository.

Merging: <https://www.atlassian.com/git/tutorials/using-branches/git-merge>

A merged version reconciling the changes from all branches to be merged is committed, and your HEAD, index, and working tree are updated to it.

Reset: <https://www.atlassian.com/git/tutorials/undoing-changes/git-reset>

The git reset command is a complex and versatile tool for undoing changes. It has three primary forms of invocation. These forms correspond to command line arguments --soft, --mixed, --hard .

Rebase: <https://www.atlassian.com/git/tutorials/rewriting-history/git-rebase>

Rebasing in Git is a process of integrating a series of commits on top of another base tip. It takes all the commits of a branch and appends them to the commits of a new branch.

Cherry-pick: <https://www.atlassian.com/git/tutorials/cherry-pick>

Cherry-picking in git means choosing a commit from one branch and applying it to another branch. This is in contrast with other ways such as merge and rebases which normally apply many commits into another branch.

Revert: <https://www.atlassian.com/git/tutorials/undoing-changes/git-revert>

The git revert command is a forward-moving undo operation that offers a safe method of undoing changes.

Tag: <https://www.atlassian.com/git/tutorials/inspecting-a-repository/git-tag>

Fork: <https://www.atlassian.com/git/tutorials/comparing-workflows/forking-workflow>

GIT Branching strategy: Trunk based Development

<https://www.atlassian.com/continuous-delivery/continuous-integration/trunk-based-development>

GIT stash: <https://www.atlassian.com/git/tutorials/saving-changes/git-stash>

Saving changes locally.

Casestudy:

* Rebase Branching strategy
* GIT revert [It’s creating additional commit]
* GIThub: <https://docs.github.com/en>