**version control :** also known as source control, is the practice of tracking and managing changes to to source code over time.Version control provides access to the historical versions of a project.

Some Source/version control management tools,

GIT

SVN

Mercurial

CVS

**Git :** is an open-source distributed version control system.

**GitHub** is a hosting service for Git repositories/projects that use Git tool.

**Features of GitHub**

-Git repositories hosting

-Project management

-Graphical representation of branches

-Collaboration

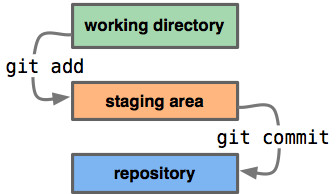
**Git Terminologies**

**Branch** - A branch is a version of the repository that diverges from the main working project

**Checkout** - used to switch between branches in a repository.

**Head** - is a reference to the last commit in the currently checked-out branch.

**Index** - index is a staging area between the working directory and repository. Which have all the files which are going to be committed.



**Master** - is a default branch of Git.

**Origin** - is a reference to the remote repository URL

**Clone** - used to make a local copy of target remote repository from the GitHub

**Fetch** - used to know the changes done in the remote repository/branch since the last pull

**Merge** - used to copy the fetched changes from remote repository/branch

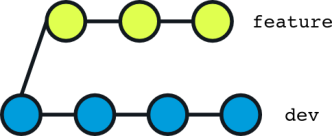
**Pull** - Fetch + Merge - it will copy the changes done in the remote repository/branch since the last pull

**Pull request** - notify the team members that they need to review the code of the feature branch and merge it into the master branch.

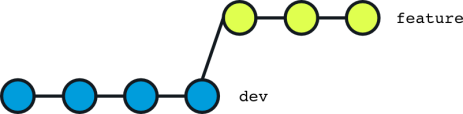
**Push** - refers to upload local repository content to a remote repository.

**Rebase** - rebasing is a technique of changing the base of your branch from one commit to another.

Before rebase



After rebase



**Stashing** - enables you to switch branch without committing the current branch.



**Steps for First time initializing git for new remote repository,**

git init (it will create .git file)

git remote add origin <https://github.com/puvinoth/TestRepo.git>

git status (gives the status of the files red- not added Green -added )

git add . (. will add all the file)

git commit -m "commit message"

git push origin master

**Steps to clone a remote repository,**

git clone <https://github.com/puvinoth/TestRepo>.git

After this just import this project in IDE

**Steps to clone a particular branch remote repository,**

git clone -b branchName <https://github.com/puvinoth/TestRepo>.git

**Steps to pull changes from remote repository after cloning,**

git pull origin master

**Steps to create new branch and push code**

git branch develop

git branch

git push https://github.com/puvinoth/TestRepo develop

git checkout develop

**Git Commands**

**Git Config command**

--local (git level access .git/config)

--global (user specific access)

--system (applies to all user in a system)

git config --global user.name "vinoth"

git config --global user.email ["vinothkumar@gmail.com"](mailto:\"vinothkumar@gmail.com\")

**Git Init command**

git init (it will create .git folder)

**Git clone command**

git clone <https://github.com/puvinoth/TestRepo>.git

**Git add command**

git add -A (stages all changes)

git add . (stages new files and modifications, without deletions)

git add -u (stages modifications and deletions, without new files)

git add filename (to add particular file)

**Git commit command**

git commit -m " Commit Message"

git commit -a (commits any files added in the repository with git add and also commits any files you've changed since then)

**Git Push command**

git push origin master (push changes to the specified branch)

git push https://github.com/puvinoth/TestRepo develop (push the created branch to the repo)

git push --all (pushes all the branches to the server repository)

**Git pull command**

git pull origin master

**Git Branch Command**

git branch develop (to create branch)

git branch (to list branches in local repo)

git branch -a (to list branches in remote repo)

git checkout develop (to switch branch)

git merge develop (to merge branch)

git branch -d develop (to delete a local branch)

git push origin -d develop (to delete a remote branch)

**Git rebase command**

git rebase develop (point the current branch to refer develop branch)

**Git fetch and merge**

git fetch origin master

git merge origin master

**Git status**

git status (gives the status of the files, whether it is added to staging or not? )