**Git link:** https://github.com/Kavya5991/repo

**Name: Kavya**

**Git branching:**

Branch is new or separate version of main repository.

Branching allows to work on different parts of project simultaneously without impacting the main branch.

**Branching strategy: GitFlow**

Gitflow enables parallel development of features where developers can create feature branch from development.

* Master
* Development
* Features
* Release
* Hotfix

Master and development branches are considered to be main branches with infinite life time and remaining branches are supporting branches usually short-lived.

**Master:** It is main branch of project that contains all history of final changes. Master branch has up – to date stable code(Production ready code). It is usually not suggested to edit the master branch of a group project, as changes would affect everyone else, and very quickly, there will be merge conflicts.

**Development:** It is parallel to the master branch. It is also considered as the main branch of the project. This branch contains the latest delivered development changes for the next release. It has the final source code for the release into dev environment.

When the develop branch reaches a stable point and is ready to release, it should be merged with master and tagged with a release version.

**Features:** **It branches from development branch and merged back to development branch**. It is mainly used to develop new features for an upcoming release.Any new implementation(specific feature) can be done by multiple developers by pulling code from developer branch and after finishing the work feature branches merged with development branch.

Example:Organisation need to implement a mobile app for shopping then home page can be considered as feature1,contact us page as feature 2 ,items page as feature 3 all these features are implemented in separate branches in feature-home,feature-contact us,feature-items and merged back to development correctly without any conflicts.

**Release:** **It branches off from development and merged back to master and development.**After code is developed and placed in development branch,release branch takes that code and performs testing in various environments like QA,Pre-production,performance and if there are any bugs developers fix the bugs from release branch and after all bugs are fixed code is ready for release and will be merged via pull request to master by assigning a new tag number. After code was successfully released in production environment, code will be merged into development branch so that future releases also contains these bug fixes.

**Hotfix:** **It branches of from master and merged back to master and development branches**. When the code is ready for release and moved to master branch from development branch and if suddenly some critical bug breaks down entire system then code should be branched off to hotfix branch from corresponding tag on master branch that marks the production version and fix those bug immediately and merge changes back to master branch.After successful release of code into production environment,code should be merged back to development branch inorder to include these bugfixes into next release.

Example:If current production release is 1.2 and a critical bug occurred then code may be branched off to hotfix 1.2.1 branch and fix the bugs then finally code merged back to master branch via pull request.

Merge to other branch can be done by raising pull request and assigning to other team members to review it and if team members approves then only code merges into other branch.

Updated codes should be present in both master and development branches as codes for next release should include previous bug fixes without any loss.