Git

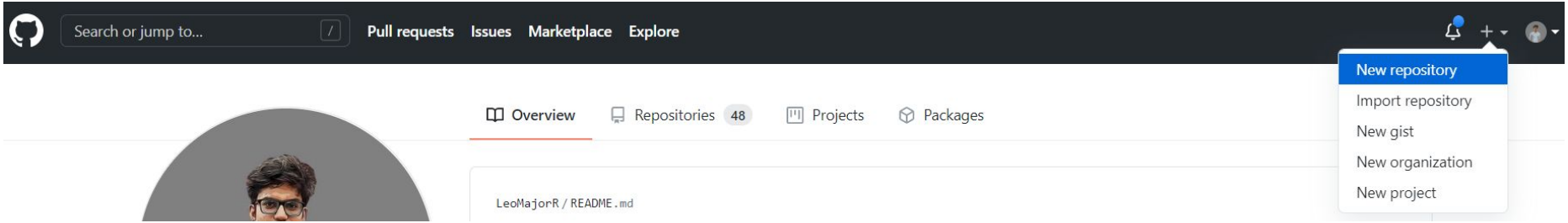
Distributed Version Control System

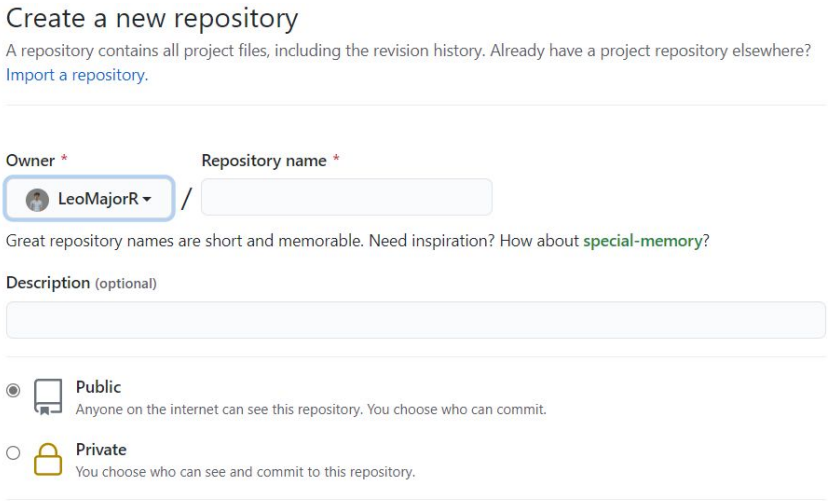
GitHub


Where the world builds software
Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world.
Email address
Sign up for GitHub
65+ million
Developers
3+ million
Organizations
200+ million
Repositories
72%
Fortune 50

GitHub

Creating a Repository





GitHub

Creating a Repository

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/LeoMajorR/GitHub-Tutorial.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# GitHub-Tutorial" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/LeoMajorR/GitHub-Tutorial.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/LeoMajorR/GitHub-Tutorial.git
git branch -M main
git push -u origin main
```

GitHub

Cloning a Repository

clone command is the most common way for users to obtain a local development clone.

```
git clone <repo url>
```

```
ssingh_raviprakash@cloudshell:~$ git clone https://github.com/LeoMajorR/GitHub-Tutorial.git
Cloning into 'GitHub-Tutorial'...
warning: You appear to have cloned an empty repository.
ssingh_raviprakash@cloudshell:~$
```

GitHub

Saving Changes to Repository

```
ssingh_raviprakash@cloudshell:~$ cd GitHub-Tutorial/
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ echo "test content for git tutorial" >> CommitTest.txt
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git add CommitTest.txt
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git commit -m "added CommitTest.txt to the repo"
[master (root-commit) 3c346ef] added CommitTest.txt to the repo
 1 file changed, 1 insertion(+)
 create mode 100644 CommitTest.txt
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

GitHub

Pushing Changes

- If you used **git clone** command to set up your local repository your repository is already configured for remote collaboration.

- Simply use git push to push your changes

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git push
Username for 'https://github.com': leomajorr
Password for 'https://leomajorr@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 275 bytes | 275.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/LeoMajorR/GitHub-Tutorial.git
 * [new branch]      master -> master
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

GitHub

Saving Credentials

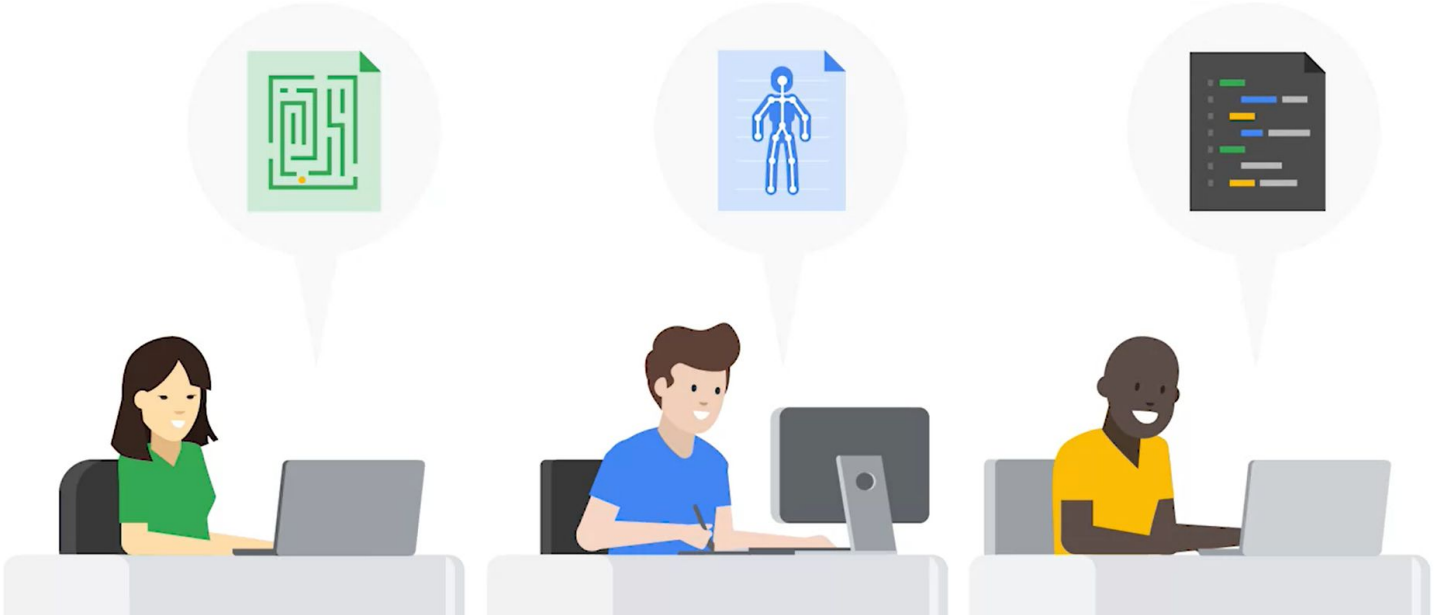
There are couple of ways to save your Credentials.

- **SSH Key-Pair :-** Create an SSH key pair and store the public key in our profile so that GitHub recognizes our computer.

- **Credential Helper :-** Credential helper caches our credentials for a time window so that we don't need to enter our password with every interaction.
  Git already comes with a credential helper baked in. We just need to enable it. We do that by calling **git config - - global credential.helper cache**.

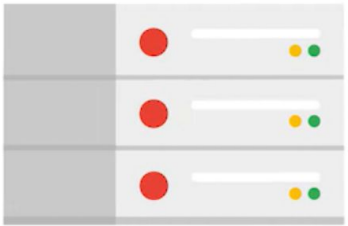
GitHub

Working with Remote



GitHub

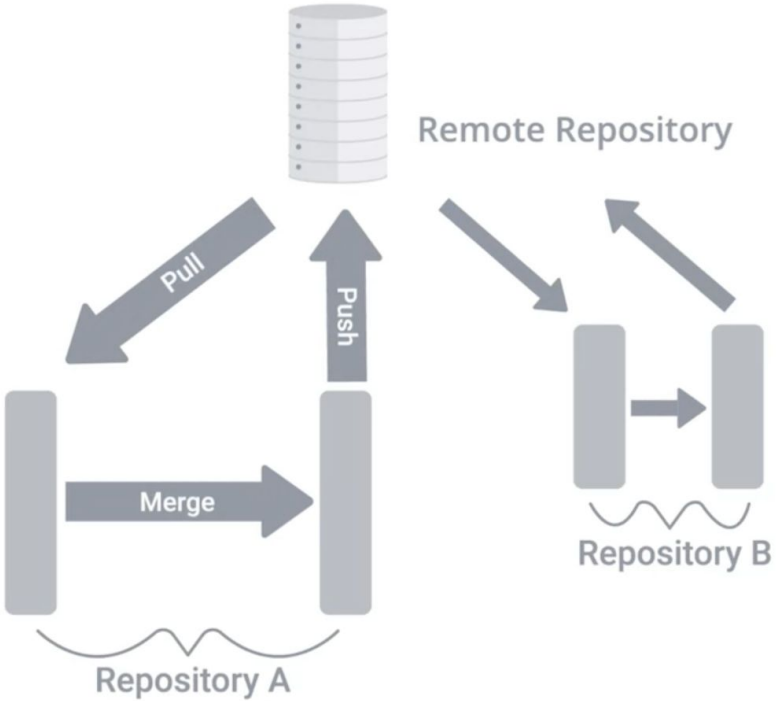
Working with Remote



Central remote repository

GitHub

Working with Remote



GitHub

Working with Remote

Git supports a variety of ways to connect to a remote repository.

- HTTP,
- HTTPS and
- SSH protocols

To look for configuration of remote

```
git remote -v
```

One will be used to fetch data from the remote repository, and the other one to push data to that remote repo.

GitHub

Working with Remote

To get more information about repo.

```
git remote show origin
```

One will be used to fetch data from the remote repository, and the other one to push data to that remote repo.

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git remote show origin
* remote origin
  Fetch URL: https://github.com/LeoMajorR/GitHub-Tutorial.git
  Push  URL: https://github.com/LeoMajorR/GitHub-Tutorial.git
  HEAD branch: master
  Remote branch:
    master tracked
  Local branch configured for 'git pull':
    master merges with remote master
  Local ref configured for 'git push':
    master pushes to master (local out of date)
```

GitHub

Fetching Changes

To Fetch all the branches from Repository

```
git fetch <remote>
```

To Fetch a specific branches from Repository

```
git fetch <remote> <branch>
```

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/LeoMajorR/GitHub-Tutorial
   3c346ef..2060600  master     -> origin/master
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

GitHub

Fetching Changes

You can look at current commit in repo using git log

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git log origin/master
commit 2060600d556eafcec25c47a08f96a1d9b081585a (origin/master)
Author: Ravi Prakash Singh <ssingh.raviprakash@gmail.com>
Date:   Wed Jun 30 20:22:33 2021 +0530

    Update CommitTest.txt

commit 3c346efacaafd7b6e58d1085863e298ca163752b (HEAD -> master)
Author: Ravi Prakash Singh <ssingh.raviprakash@gmail.com>
Date:   Sat Jun 26 10:20:46 2021 +0000

    added CommitTest.txt to the repo
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git status
On branch master
Your branch is behind 'origin/master' by 1 commit, and can be fast-forwarded.
  (use "git pull" to update your local branch)

nothing to commit, working tree clean
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

GitHub

Merging Changes

We can integrate the changes to our local repo. using `git merge`

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git merge origin/master
Updating 3c346ef..2060600
Fast-forward
 CommitTest.txt | 1 +
 1 file changed, 1 insertion(+)
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

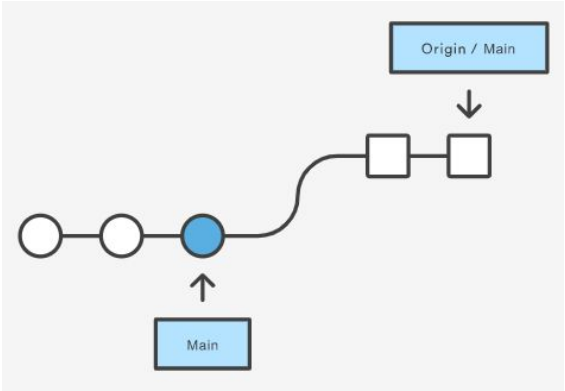
```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git log
commit 2060600d556eafcec25c47a08f96a1d9b081585a (HEAD -> master, origin/master)
Author: Ravi Prakash Singh <ssingh.raviprakash@gmail.com>
Date:   Wed Jun 30 20:22:33 2021 +0530

    Update CommitTest.txt

commit 3c346efacaafd7b6e58d1085863e298ca163752b
Author: Ravi Prakash Singh <ssingh.raviprakash@gmail.com>
Date:   Sat Jun 26 10:20:46 2021 +0000

    added CommitTest.txt to the repo
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

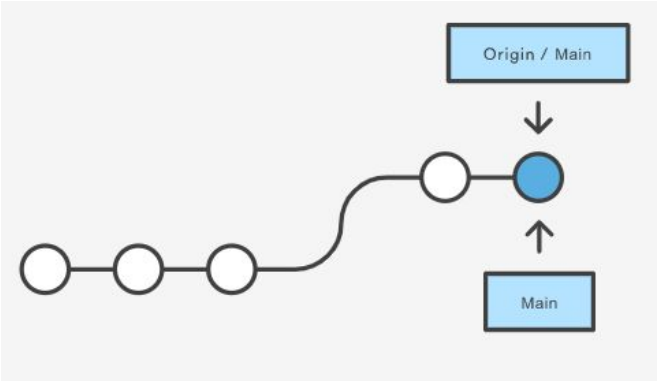
GitHub

Updating the Local Repository

Git Pull

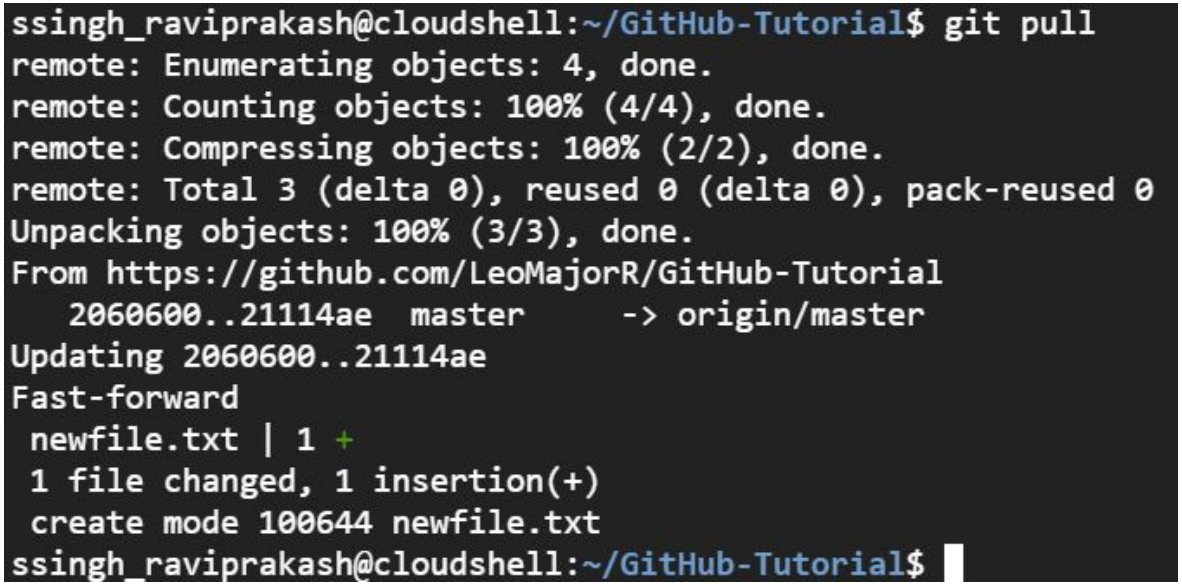
The git pull command is used to fetch and download content from a remote repository and immediately update the local repository to match that content.





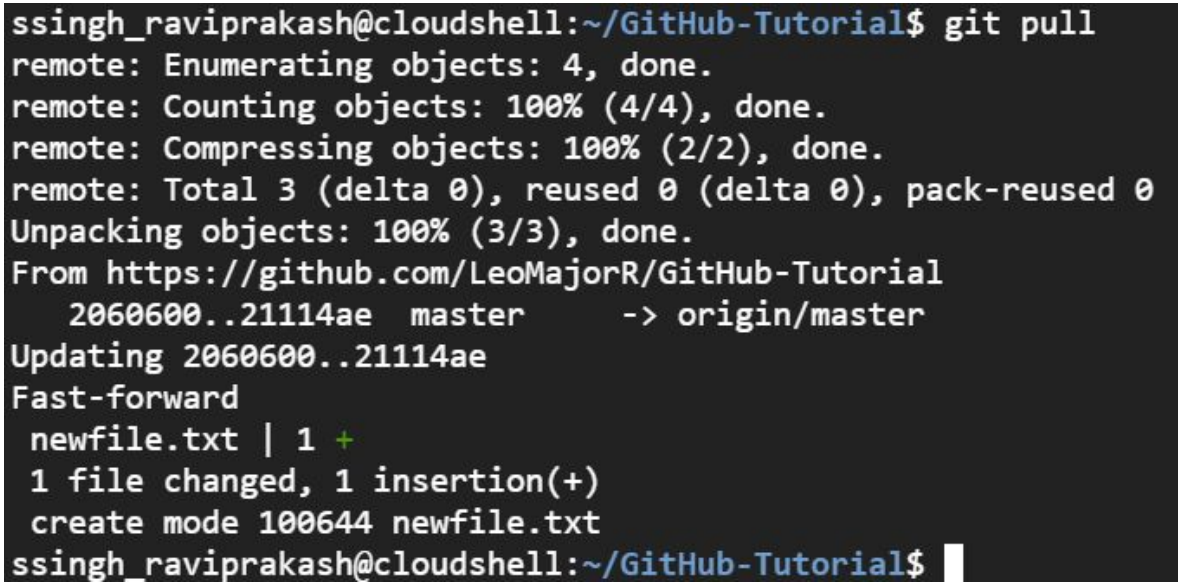
GitHub

Updating the Local Repository



GitHub

Updating the Local Repository



GitHub
Cheat Sheet

Command	Explanation
git remote	List remote repos.
git remote -v	List remote repos. verbosely
git remote show <name>	Describes a single remote repo
git remote update	Fetches most up-to-date objects
git fetch	Downloads specific Objects
git branch -r	Lists remote branches; can be combined with other branch arguments to manage remote branches

GitHub

Solving Conflicts

What if when we go to push our changes, there are new changes to the remote repo?

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git push
Username for 'https://github.com': leomajorr
Password for 'https://leomajorr@github.com':
To https://github.com/LeoMajorR/GitHub-Tutorial.git
 ! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'https://github.com/LeoMajorR/GitHub-Tutorial.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

To solve such conflict we need to perform three-way merge.

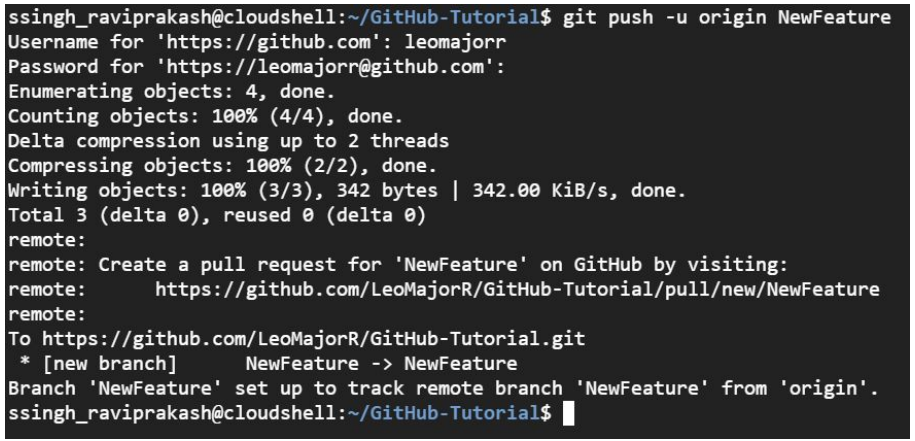
GitHub

Pushing Remote Branches

When working on new feature or a big refactor of some kind, it's recommended best practice to create separate branches.

```
git push -u origin <branch Name>
```

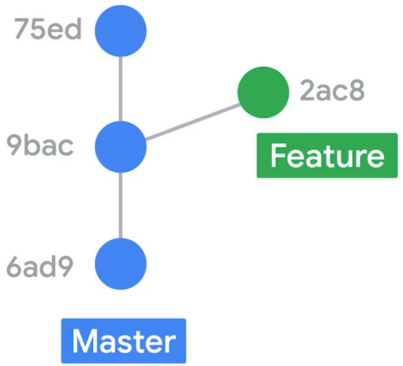
```
git push --set-upstream origin <Branch Name>
```

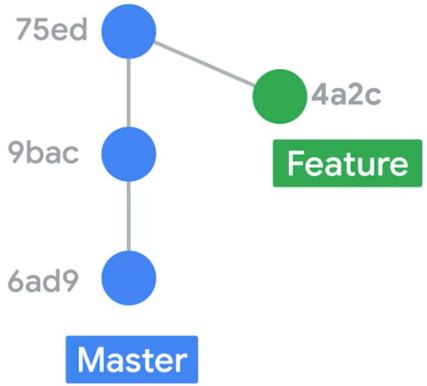


GitHub

Rebasing Changes

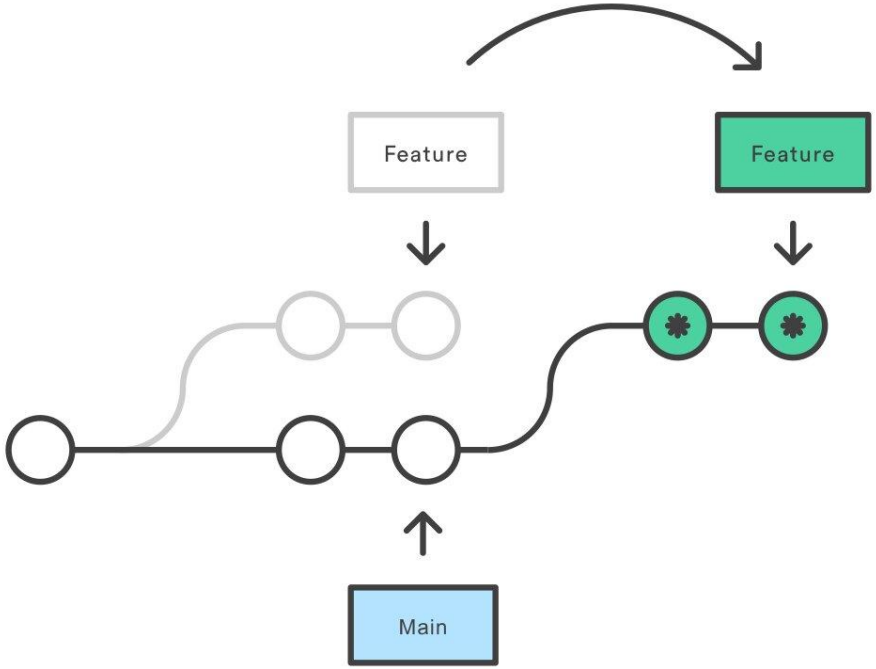
Rebasing is the process of moving or combining a sequence of commits to a new base commit.





GitHub

Rebasing Changes



GitHub

Rebasing Changes

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git log --graph --oneline --all
* 87227e3 (HEAD -> master) Notification added
| * 2dff135 (origin/NewFeature, NewFeature) Added a feature
|/
*   681aee0 (origin/master) Merge branch 'master' of https://github.com/LeoMajorR/GitHub-Tutorial New Feature fixed
|\
| * a00fc84 Update newfile.txt
* | b25aefe Added new feature
|/
* 21114ae Create newfile.txt
* 2060600 Update CommitTest.txt
* 3c346ef added CommitTest.txt to the repo
```

```
git rebase <Branch Name>
```

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git log --graph --oneline --all
* c900916 (HEAD -> NewFeature) Added a feature
* 87227e3 (master) Notification added
| * 2dff135 (origin/NewFeature) Added a feature
|/
*   681aee0 (origin/master) Merge branch 'master' of https://github.com/LeoMajorR/GitHub-Tutorial New Feature fixed
|\
| * a00fc84 Update newfile.txt
* | b25aefe Added new feature
|/
* 21114ae Create newfile.txt
* 2060600 Update CommitTest.txt
* 3c346ef added CommitTest.txt to the repo
```

GitHub

Rebasing Changes

Now Perform Fast Forward Merge

```
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$ git merge NewFeature
Updating 87227e3..c900916
Fast-forward
 Feature.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 Feature.txt
ssingh_raviprakash@cloudshell:~/GitHub-Tutorial$
```

To get rid of new branch use

```
$ git push --delete origin NewFeature
To https://github.com/LeoMajorR/GitHub-Tutorial.git
 - [deleted]         NewFeature
```

```
git push --delete origin <Branch Name>
```

GitHub

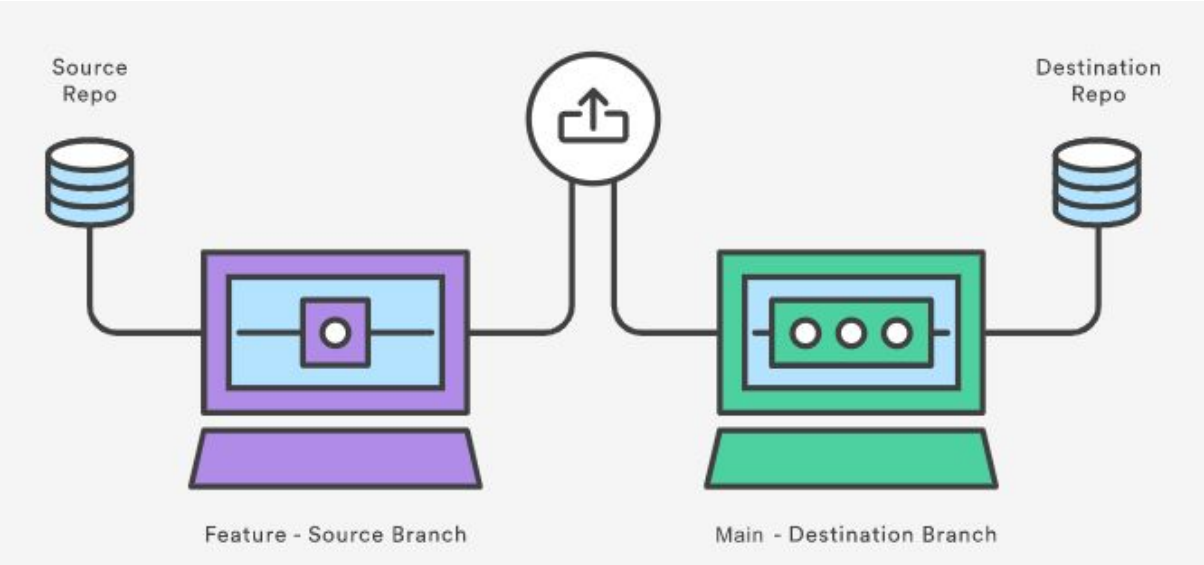
Best Practices for Collaboration

- Always synchronise your branches before starting any work of your own.
- Avoid having very large changes that modify a lot of different things.
- When working on a big change, it makes sense to have a separate feature branch.
- Regularly merge changes made on the master branch back onto the feature branch.
- Have the latest version of the project in master branch, and the stable version of the project on a separate branch.
- You shouldn't rebase changes that have been pushed to remote repos.
- Having good commit messages is important.

GitHub

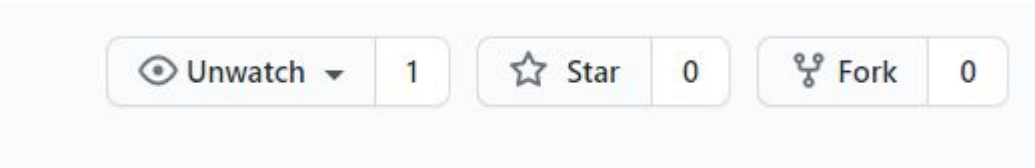
Pull Request

A Pull Request is a commit or series of commit that you send to the owner of the repository so that they incorporate it into their tree.



Forking

A way of creating a copy of the given repository so that it belongs to our user.



Forking

A way of creating a copy of the given repository so that it belongs to our user.

GitHub

Code Review

Going through someone else's code, documentation, or configuration and checking that it all makes sense and follows the expected pattern.

The goal of a code review is

- To improve the project by making sure that changes are high quality.
- It also helps us make sure that the contents are easy to understand.
- Style is consistent with the overall project. And that we don't forget any important cases.

GitHub

Style Guides

- http://google.github.io/styleguide/
- https://help.github.com/en/articles/about-pull-request-reviews
- https://medium.com/osedea/the-perfect-code-review-process-845e6ba5c31
- https://smartbear.com/learn/code-review/what-is-code-review/

Key Takepoints

- Introduction to Github
- Basic Interaction with GitHub
- Working with Remote Repository
  - Fetching new changes
  - Updating Local Repository
- Solving Conflicts
  - Pull-Merge-Push workflow
  - Rebasing your changes
- Code Review on GitHub
- Best Practices