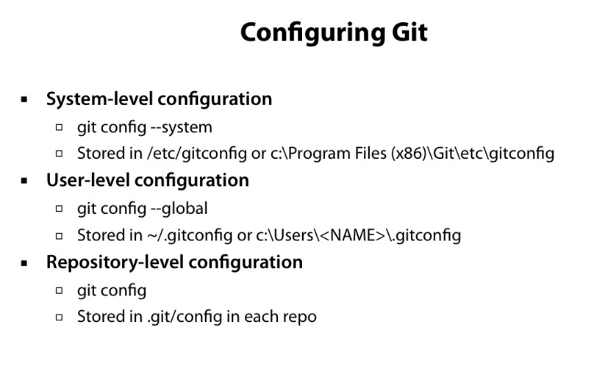
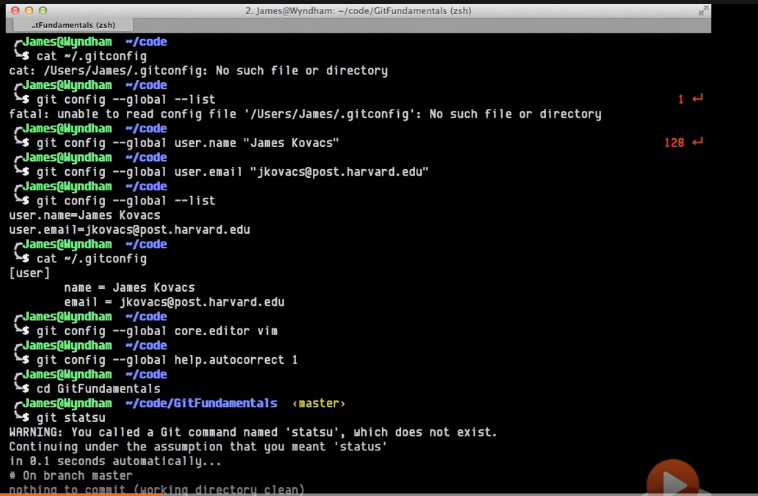
**Git – No Deep Shit**

Git is basically a distributed version control system.

For configuring git we have the below following types and commands.

It is much more common to modify the user level git configuration file.



**git config - -global - -list**

This command is used to check weather the user configurations are added or not.

If configuration are not added we can add the configurations using the below following commands.

**git config –global user.name “sahildiwan92”**

**git config –global user.email** [**isahildiwan1992@gmail.com**](mailto:isahildiwan1992@gmail.com)

We can set the core editor for our git using the below command.

**git config –global core.editor vim**

We can also install autocorrect which will basically use a fuzzy logic for maintaining the typing mistakes

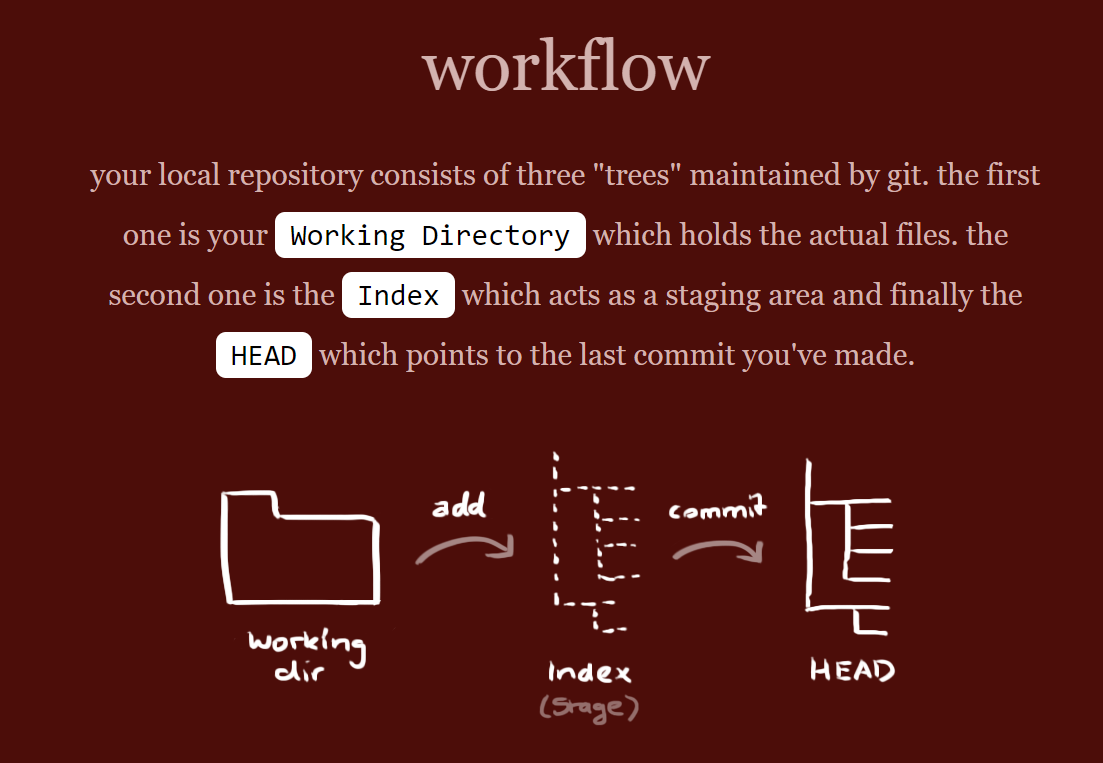
**git config –global help.autocorrect 1**

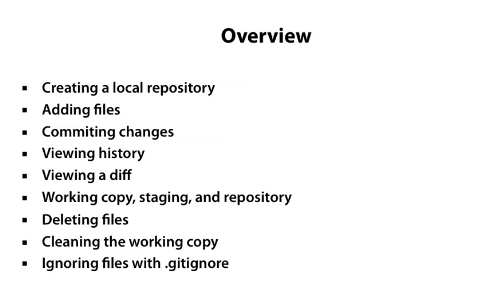
It will capture the minor mistakes.

We can add a color to the editor , to show the messages specific in a color

**git config –global color.ui auto**

**Working with Git Locally**





We can create a empty local directory by using the command

**git init**

So if a file is added we can check the added file , or the files which are changed or new data is added to it by using the command

**git status**

So the files which you have added or having changes can be added to the local git repo

**git add .** 🡪 To add all the file.

Git add specific file name to add a specific file.

**git add readme.txt**

Then we can commit the files added to the staging area.

**git commit -m “Initial commit”**

Basically the flag -m is to add the message related to the commit.

We can check the previous commits made on a local repository using the command

**git log**

X:\30th Dec 2019 - Switch\Test>git log

commit 4b9c3220724f6fd8646b12c75d393d2779809c58 (HEAD -> master)

Author: sahildiwan92 <isahildiwan1992@gmail.com>

Date: Tue Jan 21 23:08:01 2020 +0530

Initial Commit

Git identifies the commit by using SHA-1 hash of the commits.

We can add the modified files using the below command

**git add -u**

This will basically add only the updated file added not any new file.

Multiples commit will be shown in the reverse chronological order.

X:\30th Dec 2019 - Switch\Test>git log

commit 758bafbde63a67946342a4119cedae9f4ad914c4 (HEAD -> master)

Author: sahildiwan92 <isahildiwan1992@gmail.com>

Date: Tue Jan 21 23:12:17 2020 +0530

Intiial commit

commit 4b9c3220724f6fd8646b12c75d393d2779809c58

Author: sahildiwan92 <isahildiwan1992@gmail.com>

Date: Tue Jan 21 23:08:01 2020 +0530

Initial Commit

So if I want to see the change between the two git commits

**git diff 758ba.. 4b9c32**

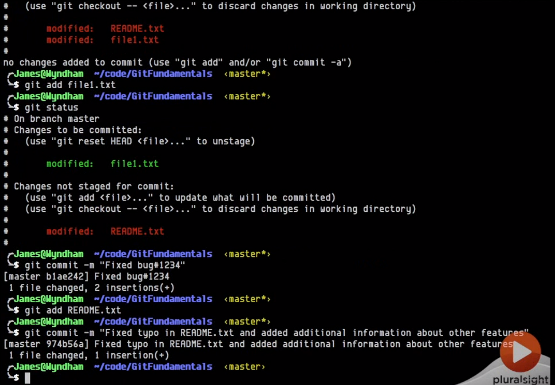
Always working with SHA is a lil difficult.

Note : The latest commit refers to as HEAD.

**git diff HEAD~1..HEAD**

~1 refers to 1 commit back from the head.

Adding different files to the different commits in git. Check the below screenshot.



Deleting a file is basically , delete from the local folder and add the changes to the git using the same commands.

Suppose I have made a change to a specific file , but I don’t want the change , we can pull the previous commit version of the file from the local HEAD version.

**git checkout filename**

**git checkout readme.txt**

We do this for the changes made by mistake.

Suppose I have did a lot of changes and I don’t want any of them we can reset the local working copy with the HEAD using the below command

**git reset –hard**

**git reset –soft HEAD~1**

Suppose if a have added a lot of files we don’t want those files then we can go for

**git clean -f**

Suppose we want to ignore some files such as the logs files

Update this for the gitingnore.