## GIT BASH

## Adding an existing project to GitHub using the command line

## Install GIT bash on machine.

## Create a new repository on GitHub. To avoid errors, do not initialize the new repository with README, license, or gitignore files. You can add these files after your project has been pushed to GitHub.

## Open Git Bash.

## Change the current working directory to your local project.

## Initialize the local directory as a Git repository. Using $ git init

## Add the files in your new local repository. This stages them for the first commit. Using $ git add .

## Commit the files that you've staged in your local repository. Using $ git commit -m "First commit"

## At the top of your GitHub repository's Quick Setup page, click to copy the remote repository URL.

## In the Command prompt, add the URL for the remote repository where your local repository will be pushed. Using $ git remote add origin ‘remote repository URL’ # Sets the new remote $ git remote -v # Verifies the new remote URL

## Push the changes in your local repository to GitHub. Using $ git push origin master Here provide the user name and password of the GIT repository.

## All Done.

## Clone Operation

## Create a new folder to clone the GIT project.

## Change the current working directory to your newly created folder.

## Now clone the project. Using

## $ git clone [gituser@git.server.com:project.git](mailto:gituser@git.server.com:project.git). Like

## $ git clone <https://github.com/loknathgupta/NODEJS.git>

## All Done.

## OTHER COMMANDS

## $ git commit -m 'Implemented my\_strlen function' (Committing Changes)

## git status -s (Seeing current status)

## git log (Seeing Logs)

## git show cbe1249b140dad24b2c35b15cc7e26a6f02d2277 (Seeing Commit Details)

## git stash

## git stash list

## git stash pop

## git pull (Pulliing updates from remote)

## git push origin master (Pushing Commits to remote)

## git checkout string\_operations.c (Revert Uncommitted Changes)

## git add string\_operations.c (Adding a new file)

## git branch new\_branch (Create Branch)

## git checkout new\_branch (Create Branch)

## git checkout new\_branch (Switch between Branches)

## git checkout -b test\_branch (\*Shortcut to Create and Switch Branch)

## git branch -D test\_branch (Delete a Branch)

## git diff (Seeing differences)

## 