**Experiment 1**

**Aim** **:** Installation of Git and creating Git Repository.

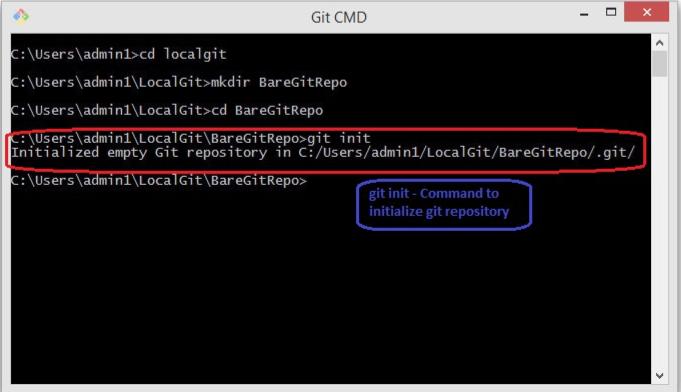
**Theory** **:** Git is a widely used open-source software tracking application used to track projects across different teams and revision level. It is a version control tool. To install Git follow the following procedure :

1. Browse to the official Git website: [https://git-scm.com/downloads](https://git-scm.com/downloads" \t "_blank)
2. Click the download link for Windows and allow the download to complete.
3. Browse to the download location (or use the download shortcut in your browser). Double-click the file to extract and launch the installer.
4. Allow the app to make changes to your device by clicking Yes on the User Account Control dialog that opens.
5. Review the GNU General Public License, and when you’re ready to install, click Next.
6. The installer will ask you for an installation location. Leave the default, unless you have reason to change it, and click Next.
7. A component selection screen will appear. Leave the defaults unless you have a specific need to change them and click Next.
8. The installer will offer to create a start menu folder. Simply click Next.
9. Select a text editor you’d like to use with Git and click Next.
10. The next step allows you to choose a different name for your initial branch. The default is ‘master.’ Unless you’re working in a team that requires a different name, leave the default option and click Next.
11. This installation step allows you to change the PATH environment. The PATH is the default set of directories included when you run a command from the command line. Leave this on the middle (recommended) selection and click Next.
12. The installer now asks which SSH client you want Git to use. Git already comes with its own SSH client, so if you don’t need a specific one, leave the default option and click Next.
13. The next option relates to server certificates. Most users should use the default. If you’re working in an Active Directory environment, you may need to switch to Windows Store certificates. Click Next.
14. The next selection converts line endings. It is recommended that you leave the default selection. This relates to the way data is formatted and changing this option may cause problems. Click Next.
15. Choose the terminal emulator you want to use. The default MinTTY is recommended, for its features. Click Next.
16. The installer now asks what the git pull command should do. The default option is recommended unless you specifically need to change its behavior. Click Next to continue with the installation.
17. Next you should choose which credential helper to use. Git uses credential helpers to fetch or save credentials. Leave the default option as it is the most stable one, and click Next.
18. Once the installation is complete, tick the boxes to view the Release Notes or Launch Git Bash, then click Finish.

**Git Repository :** Git Repository is a collection of all project files along with their history. It is a virtual storage of your project where you keep all the resources/files of the project along with a special folder called .git. The .git folder in a git repository is used by GIT programs to store information about the repository like Logs, Position of Head, and more. It allows you to save versions of your code, which can be accessed, tracked, and managed.

Steps to make git Repository :

1. Create a New Project/Folder. Command to create a folder on a Windows system is mkdir <folder name>.
2. Browse to New Project. Navigate to the project created in the previous step using the command cd <folder name>.
3. Initialize Bare Git Repository for the Project. Enter the command git init this command is used to Create Git Repository.

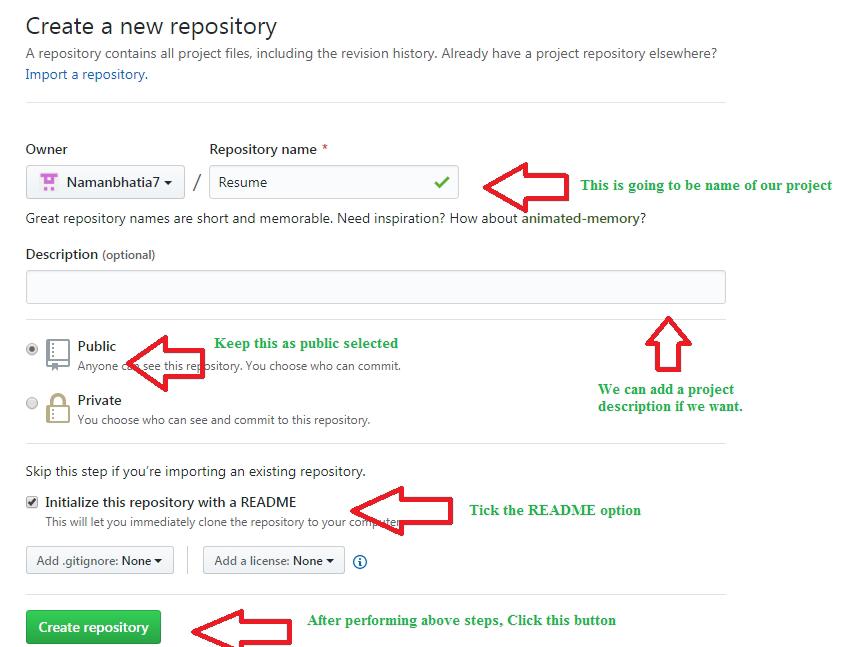


**Experiment 2**

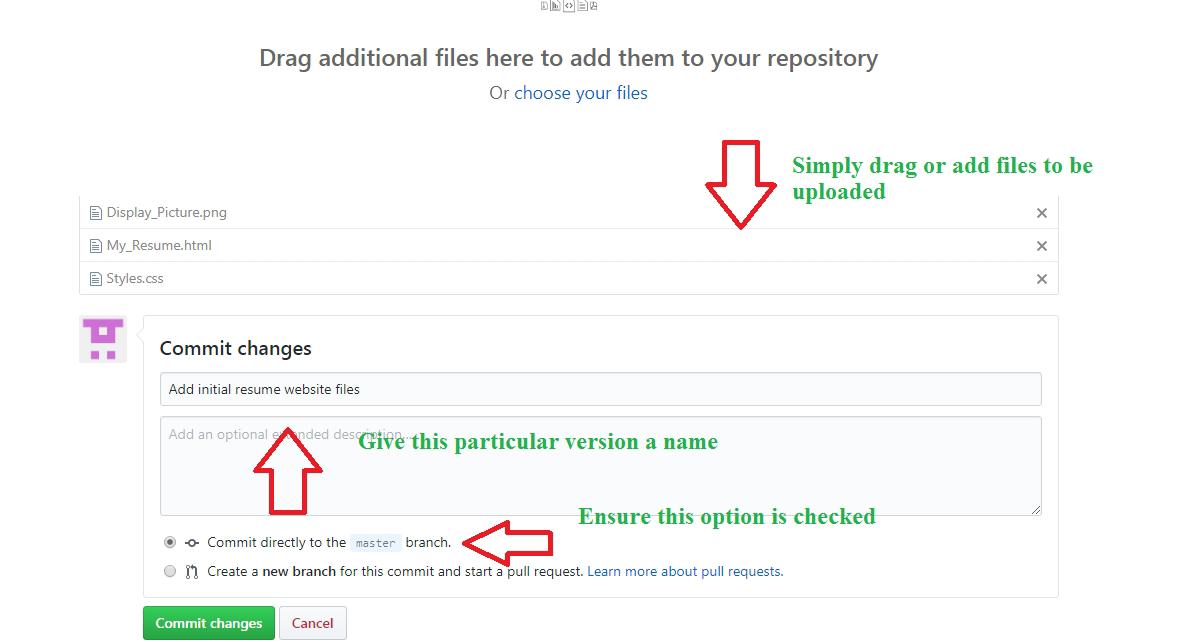
**Aim :** Creating first GitHub Repository.

**Theory :** Introduction to GitHub: Git is an open-source version control system. It means that whenever a developer develops some project (like an app) or something, he/she constantly update it catering to the demands of users, technology and whatsoever it maybe. Version control systems keep these revisions straight, storing the modifications in a central repository. It allow developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute. To create a Git Repository follow the following steps :

1. After successfully setting up GitHub account login to your account. Click on the new repository option.
2. After clicking new repository option, we will have to initialize some things like, naming our project, choosing the visibility etc. After performing these steps click Create Repository button.



1. After clicking the button, we will be directed to below page. Right now the only file we have is a readme file.
2. Now click on the “Upload files” button.
3. Follow the steps mentioned in the Picture below and click “commit changes”



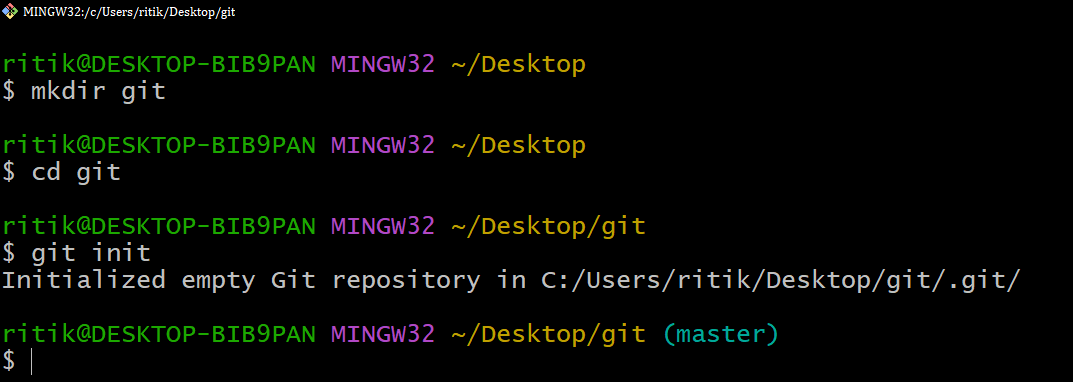
1. Now you will see that all of our files uploaded in our GitHub.

**Experiment 3**

**Aim :** Run command to initialize a repository on Git Bash.

**Theory :** The git init command creates a new Git repository. It can be used to convert an existing, unversioned project to a Git repository or initialize a new, empty repository.

1. Open Git Bash by right click and select Git Bash.
2. Make a directory by using mkdir <folder Name> in Git Bash.
3. Change the directory by using cd <folder Name>.
4. Now type git init to initialize directory as a Git Repository.



**Experiment 4**

**Aim :** Run command to add repository on Git Bash.

**Theory :**

**Experiment 5**

**Aim :** Run command to perform changes in repository on Git Bash.

**Theory :**

**Experiment 6**

**Aim :** Run command to create clone of GitHub remote repository on local system.

**Theory :**

**Experiment 7**

**Aim :** Installation of JDK and Eclipse IDE.

**Theory :**

**Experiment 8**

**Aim :** Setting up Environment Variable for JAVA.

**Theory :**

**Experiment 9**

**Aim :** Create simple JAVA program and run using CLI.

**Theory :**

**Experiment 10**

**Aim :** Create MAVEN Project in Eclipse.

**Theory :**

**Experiment 11**

**Aim :** Test MAVEN Project using Junit.

**Theory :**

**Experiment 12**

**Aim :** Installation of Docker.

**Theory :**

**Experiment 13**

**Aim :** Run various command on Docker.

**Theory :**

**Experiment 14**

**Aim :** Case study of DevOps Tool : 1) Jenkin 2) Puppet 3)Nagios.

**Theory :**

**Experiment 15**

**Aim :** Write case study on various Agile Models.

**Theory :**