

# GitHub Documentation

1. Generate file path where GitHub Repo Link will exist.  
C:\Users\mjnic\AWS\Class-7\Gitbash-GitHub
2. Familiarize yourself with the following commands:  
  
cd (change directory)- allows you to back out of or move further into your directory listings.  
pwd (print working directory)- shows the hierarchy of the directory path you are currently in.  
ls (lemme see)- shows a listing of the files in the current directory you are currently in.  
mkdir(make directory)- generates a new directory within the directory you are currently in.  
touch- creates a new file with your choice of extension in the directory you are currently in.  
rm (remove) removes the identified directory or file in the directory you're currently in.
3. Within your dedicated GitHub Repo file path on your PC, that is NOT your C Drive, navigate to the directory on your PC that will be used to establish your link with GitHub.
4. Log in to your GitHub account and we will first create a GitHub Repo through the GUI (Graphical User Interface).
5. Once logged in, Select the Repositories tab at the top left.
6. Select the green icon for a New Repository.
7. Give your new Repo a name.
8. Scroll down to the bottom and select the green icon for Create repository.
9. By following these steps you should then be automatically directed to your newly generated repo.
10. For this example I have generated the following repo "testrepo" and was returned the following url, <https://github.com/MarkofIT91/testrepo.git>
11. From accessing the url, we can upload documents to the repo through the GUI.
12. The above is how to generate and add to the repo through the GUI, now we will generate a repo from the CLI (Commnad Line Interface).
13. In the manner for this walkthrough, we will walk through the GitBash command line process for uploading documents into an existing repo.
14. "git init" establishes the connection with the existing GitHub repository in  
C:/Users/mjnic/AWS/Class-7/Gitbash-GitHub/.git/
15. "git add" stages the files that you want to add and/or modified to your existing repo on GitHub from your PC file path.
16. "git commit -m (first commit)" pushes the added and/or modified files to your GitHub repo from your PC and displays the message in parentheses.
17. "git branch -M main" changes the directory path on your PC from "Master" to "main".

18. "git remote add origin <https://github.com/MarkofIT91/GitBash-to-GitHub.git>" to specify where we want to push the updates to.
19. "git push -u origin main" to push the updates from our PC to the GitHub repo.
20. Verify through the GUI and the CLI that the updates were in fact made.

```
mjnic@LAPTOP-VQ5RN68H MINGW64 ~/aws/Class-7/Gitbash-GitHub (main)
$ git commit -m "commitments made" GitHub_Repo_Creation.md GitHub_Repo_Creation.txt README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
[main 655f6ee] commitments made
 3 files changed, 30 insertions(+)
 create mode 100644 GitHub_Repo_Creation.md
 create mode 100644 GitHub_Repo_Creation.txt

mjnic@LAPTOP-VQ5RN68H MINGW64 ~/aws/Class-7/Gitbash-GitHub (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

mjnic@LAPTOP-VQ5RN68H MINGW64 ~/aws/Class-7/Gitbash-GitHub (main)
$ git push -u origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 1.49 KiB | 1.49 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/MarkofIT91/GitBash-to-GitHub.git
   0055860..655f6ee  main -> main
branch 'main' set up to track 'origin/main'.

mjnic@LAPTOP-VQ5RN68H MINGW64 ~/aws/Class-7/Gitbash-GitHub (main)
$
```