

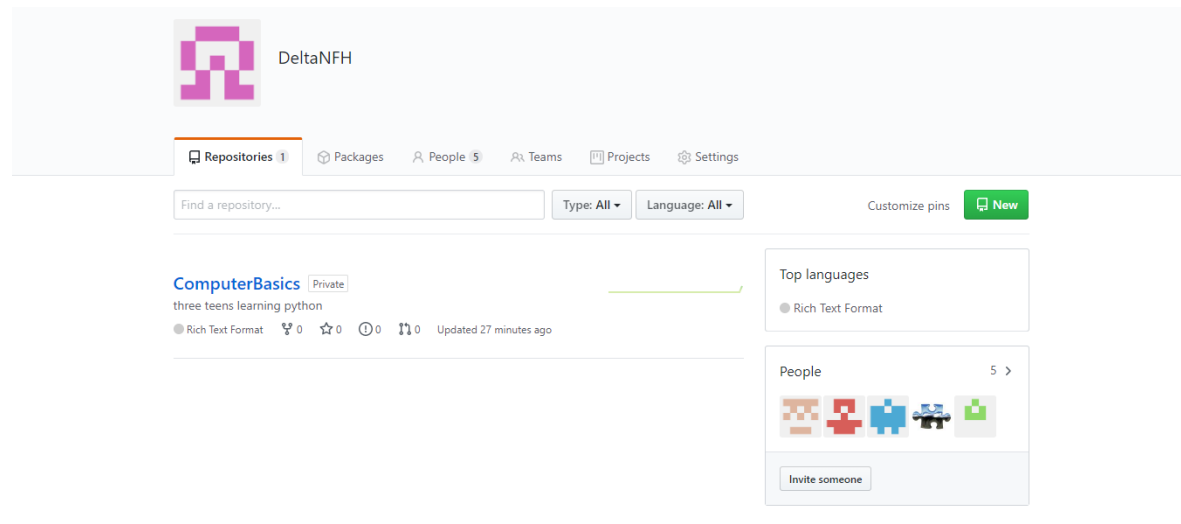
HowToGitHub101 (for Windows 10)

This document will discuss the steps needed to import your first text document to the online ComputerBasics repository.

1) How to create a GitHub Account

Use this site to <https://github.com/> create an account.

By doing so, you will be eligible to work on our repository called ComputerBasics. This is a repository (submission for documents, programs, and the like) for ProjectDeltaNFH. The submission of documents will be sent from your local computer to this online repository.



2) How to Create a Repository Directory

In your computer, create a folder called ProjectDelta. Within it, create a folder called ComputerBasics. The folder ComputerBasics will serve as your “connection” to the online ComputerBasics folder. Whatever files you import to your computer’s ComputerBasics folder, you will be able to transfer to the online ComputerBasics folder.

Whatever files you import online, the other people will be able to see, modify and evaluate which is beneficial for the project’s development.

3) How to download GitBash

Going to <https://git-scm.com/downloads> will initiate the download for the Git package. Use all the standard settings as they are chosen best. Once all the choices are selected, the installation of the app will commence.

4) How to Clone Online Repository to Computer

The first step is to direct where you want your repository to be stored in your computer. Ideally, it should be stored within the ProjectDelta folder. The code that will be used here is `cd (Directory)`. The `cd` command is intended to change the current directory.

Note - Git Bash will only work with forward slashes (/) and not with the conventional backslashes (\).

```
MINGW64/c/ProjectDelta
Fayaz_xj3hg2r@FAYAZ MINGW64 ~
$ cd C:/ProjectDelta
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta (master)
$
```

Now that we are in our desired directory, we can clone the online repository ComputerBasics to our computer. This is done with the simple command `git clone`. This command, as the name implies, clones or copies the contents to our computer. A new folder, or repository, will be created within our directory, ProjectDelta, which will be automatically named ComputerBasics.

Note – For cloning, use the url <https://github.com/DeltaNFH/ComputerBasics>. This will allow Git Bash to copy the contents within this online repository to our computer-repository.

```
MINGW64/c/ProjectDelta
Fayaz_xj3hg2r@FAYAZ MINGW64 ~
$ cd C:/ProjectDelta
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta (master)
$ git clone https://github.com/DeltaNFH/ComputerBasics
Cloning into 'ComputerBasics'...
remote: Enumerating objects: 54, done.
remote: Counting objects: 100% (54/54), done.
remote: Compressing objects: 100% (40/40), done.
remote: Total 54 (delta 21), reused 31 (delta 9), pack-reused 0
Unpacking objects: 100% (54/54), 17.65 KiB | 67.00 KiB/s, done.
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta (master)
$
```

5) How to create and edit a text file (Notepad)

Creating a text file can be done easily with one step. Use the command `touch` (file name and type). This will create within the repository a new file specified by the name and type provided by us. The file size will always be 0 bytes.

Note – We have to first change the directory to the repository by once doing the `cd` command.

```
MINGW64/c/ProjectDelta/ComputerBasics
Fayaz_xj3hg2r@FAYAZ MINGW64 ~
$ cd C:/ProjectDelta
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta (master)
$ git clone https://github.com/DeltaNFH/ComputerBasics
Cloning into 'ComputerBasics'...
remote: Enumerating objects: 54, done.
remote: Counting objects: 100% (54/54), done.
remote: Compressing objects: 100% (40/40), done.
remote: Total 54 (delta 21), reused 31 (delta 9), pack-reused 0
Unpacking objects: 100% (54/54), 17.65 KiB | 67.00 KiB/s, done.
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta (master)
$ cd C:/ProjectDelta/ComputerBasics
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ touch Testing.txt
Fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$
```

To edit your newly created-text file, use the command `vi` (file name). It's important to note that `vi` only works for text documents, which includes Notepad files. A `vi` editor will pop up which will allow you to edit the file. First type capital `I` then start typing. When you are done, press `esc` and type `:wq`. Typing `w` will save and `q` will exit.


```
MINGW64~/c/ProjectDelta/ComputerBasics
$ git clone https://github.com/DeltaNFH/ComputerBasics
Cloning into 'ComputerBasics'...
remote: Enumerating objects: 54, done.
remote: Counting objects: 100% (54/54), done.
remote: Compressing objects: 100% (40/40), done.
remote: Total 54 (delta 21), reused 31 (delta 9), pack-reused 0
Unpacking objects: 100% (54/54), 17.65 KiB | 45.00 KiB/s, done.

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ cd C:/ProjectDelta/ComputerBasics

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ touch Testing.txt

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ vi Testing.txt

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ git add Testing.txt
warning: LF will be replaced by CRLF in Testing.txt.
The file will have its original line endings in your working directory

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ git commit Testing.txt
```

```
MINGW64~/c/ProjectDelta/ComputerBasics
This is a sample text document.
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is up to date with 'origin/master'.
#
# Changes to be committed:
#   new file:   Testing.txt
#
# Changes not staged for commit:
#   modified:   HELLO.txt
#
# Untracked files:
#   Arshad.docx
#   LOL.txt
#
~
~
~
~/c/ProjectDelta/ComputerBasics/.git/COMMIT_EDITMSG(+) [unix] (20:35 06/06/2020)1,31 All
~
~
```

The last step is finally submitting the text document to the online repository. This is done by `git push`. That's it!

You should be seeing your text document when you now log in to the online GitHub repository.

Note – It's safe to first check by using `git pull`. If something changes to the repository while doing your `git push` command, then your attempt will be rejected. `git pull` will essentially update your repository for any changes that may have accorded. Then it is safe to `git push`.

```
MINGW64~/c/ProjectDelta/ComputerBasics
fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ git push
To https://github.com/DeltaNFH/ComputerBasics
 ! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'https://github.com/DeltaNFH/ComputerBasics'
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.

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ git pull
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 2 (delta 1), reused 1 (delta 1), pack-reused 0
Unpacking objects: 100% (2/2), 599 bytes | 37.00 KiB/s, done.
From https://github.com/DeltaNFH/ComputerBasics
 9f8aa02..369bfe1 master -> origin/master
Updating 9f8aa02..369bfe1
Fast-forward
 LOL.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 LOL.txt

fayaz_xj3hg2r@FAYAZ MINGW64 /c/ProjectDelta/ComputerBasics (master)
$ git push
Everything up-to-date
```