

Preliminary Task – 1

Git and Github

Submitted By :

Melvina Jose

1st year CYS

Creating a GitHub account

Installing Git In Your desktop

Git Operators

Configuring Git Bash

```
Me1@Me1 MINGW64 ~
$ git --version
git version 2.35.1.windows.2

Me1@Me1 MINGW64 ~
$ git config --global user.name melvinajose

Me1@Me1 MINGW64 ~
$ git config --global user.email mel.melvinajose@gmail.com

Me1@Me1 MINGW64 ~
$ git config --global --list
user.name=melvinajose
user.email=mel.melvinajose@gmail.com
```

Creating “Cognizance” Repo using Git Bash - init

```
Me1@Me1 MINGW64 ~
$ cd cognizance

Me1@Me1 MINGW64 ~/cognizance
$ git init
Initialized empty Git repository in C:/Users/me1me/cognizance/.git/

Me1@Me1 MINGW64 ~/cognizance (master)
$ ls

Me1@Me1 MINGW64 ~/cognizance (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

git add

adds a change in the working directory to the staging area.

```

Me1@Me1 MINGW64 ~/cognizance (master)
$ touch Task-1

Me1@Me1 MINGW64 ~/cognizance (master)
$ ls -l
total 0
-rw-r--r-- 1 Me1 197609 0 Feb 14 21:03 Task-1

Me1@Me1 MINGW64 ~/cognizance (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
      Task-1

nothing added to commit but untracked files present (use "git add" to track)

Me1@Me1 MINGW64 ~/cognizance (master)
$ git add Task-1

Me1@Me1 MINGW64 ~/cognizance (master)
$ git ls-files
Task-1

Me1@Me1 MINGW64 ~/cognizance (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
      new file:   Task-1

```

git commit - captures a snapshot of the project's currently staged changes.

git log - review and read a history of everything that happens to a repository.

```

Me1@Me1 MINGW64 ~/cognizance (master)
$ git commit -m "Created a folder Task-1"
[master (root-commit) 370d7c1] Created a folder Task-1
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Task-1

Me1@Me1 MINGW64 ~/cognizance (master)
$ git log
commit 370d7c1cea02046a78d1912ed9575fffb1845e8 (HEAD -> master)
Author: melvinajose <mel.melvinajose@gmail.com>
Date:   Mon Feb 14 21:06:17 2022 +0530

    Created a folder Task-1

```

git remote - manages the set of remotes that you are tracking with your local repository.

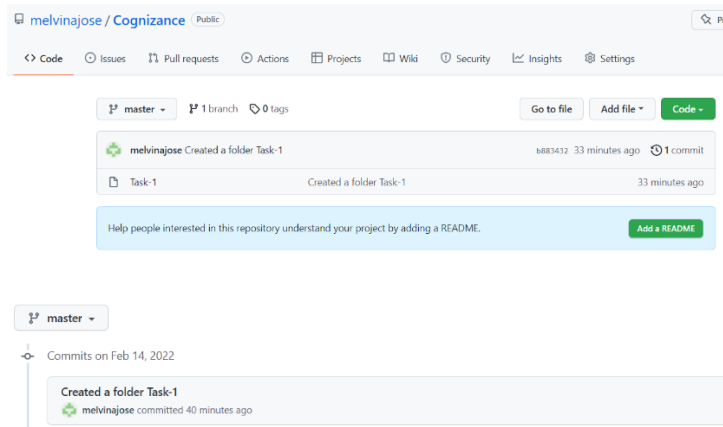
git push - used to upload local repository content to a remote repository.

```

Me1@Me1 MINGW64 ~/cognizance (master)
$ git remote add origin https://github.com/melvinajose/Cognizance.git

Me1@Me1 MINGW64 ~/cognizance (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 220 bytes | 220.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/melvinajose/Cognizance/pull/new/master
remote:
To https://github.com/melvinajose/Cognizance.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

```



git branch- lets you create, list, rename, and delete branches.

```
Me1@Me1 MINGW64 ~/cognizance (master)
$ git branch -M branch1

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ ls
Task-1
```

git checkout - lets you navigate between the branches created by git branch .

git status- displays the state of the working directory and the staging area.

```
Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git checkout master
Switched to a new branch 'master'
branch 'master' set up to track 'origin/master'.

Me1@Me1 MINGW64 ~/cognizance (master)
$ git push
Everything up-to-date

Me1@Me1 MINGW64 ~/cognizance (master)
$ git commit -m "Created a branch Branch1"
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

Me1@Me1 MINGW64 ~/cognizance (master)
$ git push -u origin master
Everything up-to-date
branch 'master' set up to track 'origin/master'.

Me1@Me1 MINGW64 ~/cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

Me1@Me1 MINGW64 ~/cognizance (master)
$ git add .

Me1@Me1 MINGW64 ~/cognizance (master)
$ git ls-files
Task-1

Me1@Me1 MINGW64 ~/cognizance (master)
$ git push
Everything up-to-date
```

```

Me1@Me1 MINGW64 ~/cognizance (master)
$ git checkout branch1
Switched to branch 'branch1'
Your branch is up to date with 'origin/master'.

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git push -u origin branch1
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'branch1' on GitHub by visiting:
remote:   https://github.com/melvinajose/Cognizance/pull/new/branch1
remote:
To https://github.com/melvinajose/Cognizance.git
 * [new branch]      branch1 -> branch1
warning: branch.branch1.remote has multiple values
warning: branch.branch1.merge has multiple values
branch 'branch1' set up to track 'origin/branch1'.

```

The screenshot shows the GitHub interface for the repository 'melvinajose/Cognizance'. The 'Overview' tab is selected, displaying a search bar for branches. Under 'Default branch', 'master' is listed as the default. Under 'Your branches', 'branch1' is shown, updated 3 hours ago, with a 'New pull request' button. Under 'Active branches', 'branch1' is also shown with a 'New pull request' button. The repository is public and has 1 unwatch, 0 forks, and 0 stars.

git pull - used to update the local version of a repository from a remote.

git ls-files - outputs the filenames

The screenshot shows the GitHub interface for the repository 'melvinajose/Cognizance' on the 'branch1' branch. It indicates that the branch is 1 commit ahead of master. A commit by 'melvinajose' is shown, which created a new file 'new.txt' in branch1. The commit message is 'Created new.txt in branch1'.

```

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git ls-files
Task-1

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git pull origin branch1
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 677 bytes | 96.00 KiB/s, done.
From https://github.com/melvinajose/Cognizance
 * branch            branch1      -> FETCH_HEAD
 * b883432..06d19ea  branch1      -> origin/branch1
Updating b883432..06d19ea
Fast-forward
 new.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 new.txt

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git ls-files
Task-1
new.txt

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ cat new.txt
My first git repo.

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git status
On branch branch1
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean

```

```

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git log
commit 06d19eaf8c863c32acdf139375066b1314bb8aac (HEAD -> branch1, origin/branch1)
Author: melvinajose <99662316+melvinajose@users.noreply.github.com>
Date: Tue Feb 15 00:34:25 2022 +0530

    Created new.txt in branch1

commit b88343248dad059ff5c45aa32f0bb6547d73f68a (origin/master, master)
Author: melvinajose <me1.melvinajose@gmail.com>
Date: Mon Feb 14 21:42:13 2022 +0530

    Created a folder Task-1

```

git merge - used to integrate changes from another branch.

```

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git branch -M merge_branch

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ touch merge

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ git add .

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ git ls-files
Task-1
merge
new.txt

```

```

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git branch -M merge_branch

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ touch merge

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ git add .

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ git ls-files
Task-1
merge
new.txt

Me1@Me1 MINGW64 ~/cognizance (merge_branch)
$ git checkout branch1
Switched to a new branch 'branch1'
A      merge
branch 'branch1' set up to track 'origin/branch1'.

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git merge merge_branch
Already up to date.

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git ls-files
Task-1
merge
new.txt

Me1@Me1 MINGW64 ~/cognizance (branch1)
$ git commit -m "Merged branch1 with merge_branch"
[branch1 2896781] Merged branch1 with merge_branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 merge

```

Default branch			
master	Updated 5 hours ago by melvinajose	Default	
Your branches			
merge_branch	Updated 2 hours ago by melvinajose	0/2	New pull request edit delete
branch1	Updated 2 hours ago by melvinajose	0/1	New pull request edit delete
Active branches			
merge_branch	Updated 2 hours ago by melvinajose	0/2	New pull request edit delete
branch1	Updated 2 hours ago by melvinajose	0/1	New pull request edit delete

git clone - used to create a copy of a specific repository or branch within a repository.

```
Me1@Me1 MINGW64 ~/cognizance (master)
$ mkdir clone

Me1@Me1 MINGW64 ~/cognizance (master)
$ cd

Me1@Me1 MINGW64 ~
$ cd cognizance/clone/

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git clone https://github.com/melvinajose/Cognizance.git
Cloning into 'Cognizance'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 8 (delta 1), reused 5 (delta 0), pack-reused 0
Receiving objects: 100% (8/8), done.
Resolving deltas: 100% (1/1), done.

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ ls
Cognizance/

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ cd cognizance/

Me1@Me1 MINGW64 ~/cognizance/clone/cognizance (master)
$ ls
Task-1
```

git rebase - used to change a series of commits, modifying the history of your repository.

```
Me1@Me1 MINGW64 ~/cognizance (master)
$ cd clone/

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git rebase master
Current branch master is up to date.

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git branch rebase

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git checkout rebase
Switched to branch 'rebase'

Me1@Me1 MINGW64 ~/cognizance/clone (rebase)
$ ls
Cognizance/

Me1@Me1 MINGW64 ~/cognizance/clone (rebase)
$ cd cognizance/
bash: cognizance/: Is a directory

Me1@Me1 MINGW64 ~/cognizance/clone (rebase)
$ cd cognizance

Me1@Me1 MINGW64 ~/cognizance/clone/cognizance (master)
$ touch rebase

Me1@Me1 MINGW64 ~/cognizance/clone/cognizance (master)
$ git add clone
fatal: pathspec 'clone' did not match any files

Me1@Me1 MINGW64 ~/cognizance/clone/cognizance (master)
$ git add rebase

Me1@Me1 MINGW64 ~/cognizance/clone/cognizance (master)
$ git commit -m "rebase"
[master 0e66f37] rebase
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 rebase

Me1@Me1 MINGW64 ~/cognizance/clone/cognizance (master)
$ git rebase master
Current branch master is up to date.
```

git fetch - downloads commits, files, and refs from a remote repository into your local repo.

```
Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git fetch https://github.com/melvinajose/Cognizance.git master
From https://github.com/melvinajose/Cognizance
* branch          master       -> FETCH_HEAD

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git fetch -all
error: did you mean '--all' (with two dashes)?

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git fetch --all
Fetching origin
```

git reset - used to undo local changes to the state of a Git repo.

```
Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ touch reset
a
Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git add reset

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git reset --mixed

Me1@Me1 MINGW64 ~/cognizance/clone (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  ./

nothing added to commit but untracked files present (use "git add" to track)
```

git stash - saves the uncommitted changes locally, allowing you to make changes, switch branches, and perform other Git operations.

```
Me1@Me1 MINGW64 ~/cognizance (master)
$ git stash
No local changes to save
```

git fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.

melvinajose / google-ctf Public
forked from google/google-ctf

<> Code Pull requests Actions Projects Security Insights Settings

master 1 branch 0 tags

This branch is up to date with google/master. Contribute Fetch upstream

matrizzo Add the resources for Hackceler8 2021. b5ba474 26 days ago 331 commits

2017	Remove double hashbang	4 years ago
2018	open source pwn-mojo from finals 2018	3 years ago
2019	Update main.py	13 months ago
2020	Adding Hackceler8 2020 resources	8 months ago
2021	Add the resources for Hackceler8 2021.	26 days ago
infrastructure	Update index.html	13 months ago
other/re-risky	Adding an old guest challenge made for Winja CTF	2 years ago
third_party	Add the resources for Hackceler8 2021.	26 days ago

About

Google CTF

[g.co/ctf](#)

Readme

Apache-2.0 License

0 stars

0 watching

403 forks

Releases

No releases published

Create a new release

Packages

No packages published

google-ctf

Public

Forked from google/google-ctf

Google CTF

Go

git config --list --show-origin

```
Me1@Me1 MINGW64 ~/cognizance (master)
$ git config --list --show-origin
file:C:/Program Files/Git/etc/gitconfig diff.astextplain.textconv=astextplain
file:C:/Program Files/Git/etc/gitconfig filter.lfs.clean=git-lfs clean -- %f
file:C:/Program Files/Git/etc/gitconfig filter.lfs.smudge=git-lfs smudge -- %f
file:C:/Program Files/Git/etc/gitconfig filter.lfs.process=git-lfs filter-process
file:C:/Program Files/Git/etc/gitconfig filter.lfs.required=true
file:C:/Program Files/Git/etc/gitconfig http.sslbackend=openssl
file:C:/Program Files/Git/etc/gitconfig http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
file:C:/Program Files/Git/etc/gitconfig core.autocrlf=true
file:C:/Program Files/Git/etc/gitconfig core.fscache=true
file:C:/Program Files/Git/etc/gitconfig core.symlinks=false
file:C:/Program Files/Git/etc/gitconfig pull.rebase=false
file:C:/Program Files/Git/etc/gitconfig credential.helper=manager-core
file:C:/Program Files/Git/etc/gitconfig credential.https://dev.azure.com.usehttppath=true
file:C:/Program Files/Git/etc/gitconfig init.defaultbranch=master
file:C:/Users/me1/.gitconfig user.name=melvinajose
file:C:/Users/me1/.gitconfig user.email=mel.melvinajose@gmail.com
file:./git/config core.repositoryformatversion=0
file:./git/config core.filemode=false
file:./git/config core.bare=false
file:./git/config core.logallrefupdates=true
file:./git/config core.symlinks=false
file:./git/config core.ignorecase=true
file:./git/config remote.origin.url=https://github.com/melvinajose/Cognizance.git
file:./git/config remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
file:./git/config branch.master.remote=origin
file:./git/config branch.master.merge=refs/heads/master
file:./git/config branch.branch1.remote=origin
file:./git/config branch.branch1.merge=refs/heads/branch1
```

git config --list --show-scope

```
Me1@Me1 MINGW64 ~/cognizance (master)
$ git config --list --show-scope
system diff.astextplain.textconv=astextplain
system filter.lfs.clean=git-lfs clean -- %f
system filter.lfs.smudge=git-lfs smudge -- %f
system filter.lfs.process=git-lfs filter-process
system filter.lfs.required=true
system http.sslbackend=openssl
system http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
system core.autocrlf=true
system core.fscache=true
system core.symlinks=false
system pull.rebase=false
system credential.helper=manager-core
system credential.https://dev.azure.com.usehttppath=true
system init.defaultbranch=master
global user.name=melvinajose
global user.email=mel.melvinajose@gmail.com
local core.repositoryformatversion=0
local core.filemode=false
local core.bare=false
local core.logallrefupdates=true
local core.symlinks=false
local core.ignorecase=true
local remote.origin.url=https://github.com/melvinajose/Cognizance.git
local remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
local branch.master.remote=origin
local branch.master.merge=refs/heads/master
local branch.branch1.remote=origin
local branch.branch1.merge=refs/heads/branch1
```

X-----X