

NAME – CHALLA SHIVA SHANKAR KOUSHIK REDDY

ROLLNO – CH.EN.U4CSE20015

DISCORD HANDLE -KOUSHIK#4875

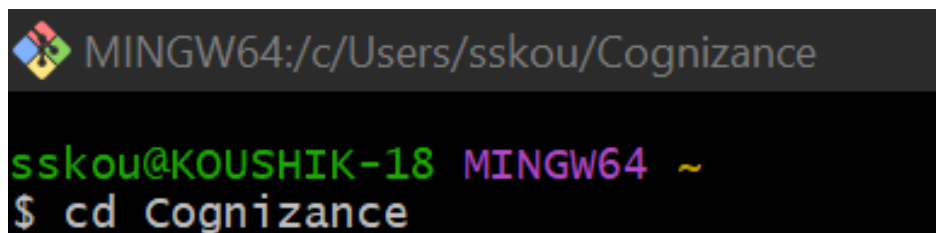
TASK 1 [GIT & GITHUB]

#Creating An GitHub Account

- I Already have my GitHub account.
- My GitHub Account Link - <https://github.com/sskoushik>

#Install Git In Your Desktop


- GitHub Is Already Installed On My Laptop.



```
MINGW64:/c/Users/sskou/Cognizance  
sskou@KOUSHIK-18 MINGW64 ~  
$ cd Cognizance
```

1.create a repository with name "Cognizance" on GitHub using Git Bash.

Owner *

 sskoushik ▾

/


Repository name *

Cognizance ✓


Great repository names are short and memorable. Need inspiration? How about [turbo-lamp?](#)

Description (optional)

Cognizance (Task)

☒  **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore



Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: **None** ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Quick setup — if you've done this kind of thing before

 Set up in Desktop or **HTTPS** **SSH** 

Get started by creating a new file or uploading an existing file. We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line


```
echo "# Cognizance" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/sskoushik/Cognizance.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/sskoushik/Cognizance.git
git branch -M main
git push -u origin main
```

sskoushik / Cognizance Public

 Pin

 Unwatch

<> Code


 Issues


 Pull requests

 Actions


 Projects

 Wiki

 Security

 Insights

 Settings

 main ▾

 1 branch

 0 tags

Go to file


Add file ▾

Code ▾



sskoushik first commit

a1e6eeb 17 hours ago

 1 commit



TEXT.txt

first commit

17 hours ago

Help people interested in this repository understand your project by adding a README.

Add a README

```
sskou@KOUSHIK-18 MINGW64 ~  
$ mkdir Cognizance  
  
sskou@KOUSHIK-18 MINGW64 ~  
$ cd Cognizance  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance  
$ git init  
Initialized empty Git repository in C:/Users/sskou/Cognizance/.git/  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)  
$ mkdir task1  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)  
$ cd task1  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance/task1 (master)  
$ touch Text.txt  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance/task1 (master)  
$ cd ..  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)  
$ git init  
Reinitialized existing Git repository in C:/Users/sskou/Cognizance/.git/  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)  
$ git add *  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)  
$ git status  
On branch master  
  
No commits yet  
  
Changes to be committed:  
  (use "git rm --cached <file>..." to unstage)  
    new file:   task1/Text.txt  
  
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)  
$ git commit -m "first commit"  
[master (root-commit) 6bc54a8] first commit  
 1 file changed, 0 insertions(+), 0 deletions(-)  
 create mode 100644 task1/Text.txt
```

2.Create a folder with name "Task-1" in the repository and perform the Git operations using Git Bash.

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)
$ mkdir task1
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)
$ cd task1
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)
$ git commit -m "first commit"
[master (root-commit) 6bc54a8] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 task1/Text.txt
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)
$ git branch -M main
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git remote add origin |
usage: git remote add [<options>] <name> <url>
```

```
    -f, --fetch                fetch the remote branches
    --tags                    import all tags and associated objects when fetching
                              or do not fetch any tag at all (--no-tags)
    -t, --track <branch>     branch(es) to track
    -m, --master <branch>    master branch
    --mirror[=(push|fetch)]  set up remote as a mirror to push to or fetch from
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git remote add origin https://github.com/sskoushik/Cognizance.git
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (4/4), 252 bytes | 126.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/sskoushik/Cognizance.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git pull
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), 673 bytes | 56.00 KiB/s, done.
From https://github.com/sskoushik/Cognizance
   6bc54a8..588715e  main      -> origin/main
Updating 6bc54a8..588715e
Fast-forward
 task1/Text.txt | 1 +
1 file changed, 1 insertion(+)
```

3.Upload the screenshot of the git bash using This command in your computer.

```
git config --list --show-origin
```

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ 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/sskou/.gitconfig user.email=sskoushik.challa@gmail.com
file:C:/Users/sskou/.gitconfig user.name=sskoushik
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/sskoushik/Cognizance.git
file:.git/config remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
file:.git/config branch.main.remote=origin
file:.git/config branch.main.merge=refs/heads/main
```

4. Take a screenshot of all the Git operations you performed and compile it in a single PDF and upload it in "Task-1" folder this PDF Should Also Contain the screenshot of the command mentioned in point 3.

Basic Operations

The following is a summary of basic git operations:

Git Add:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (master)
$ git add *
```

Git Checkout :

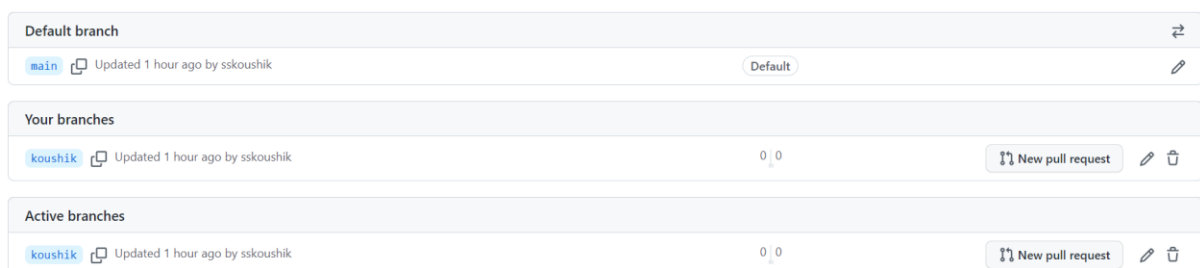
Replaces the current working files with files from a branch.

Git Checkout - B:

Creates a new local branch from the current branch's tip.

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git checkout -b koushik
Switched to a new branch 'koushik'

sskou@KOUSHIK-18 MINGW64 ~/Cognizance (koushik)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```



Git Push:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git push origin koushik
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'koushik' on GitHub by visiting:
remote:   https://github.com/sskoushik/Cognizance/pull/new/koushik
remote:
To https://github.com/sskoushik/Cognizance.git
 * [new branch]      koushik -> koushik
```

Git Pull:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git pull
Already up to date.
```

Git Commit:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git commit
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

Git Commit -a:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git commit -a
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

Git Fetch:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git fetch
```

Git Merge:


```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git merge koushik main
Already up to date.
```

Git Clone:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git clone https://github.com/sskoushik/Cognizance.git
Cloning into 'Cognizance'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 8 (delta 0), reused 4 (delta 0), pack-reused 0
Receiving objects: 100% (8/8), done.
```

Git Reset:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git reset
```

Git Rebase:

```
sskou@KOUSHIK-18 MINGW64 ~/Cognizance (main)
$ git rebase main
Current branch main is up to date.
```

Git Fork:

Fork will create a copy of the repository in your GitHub account so that you can make changes to the project

