

are View

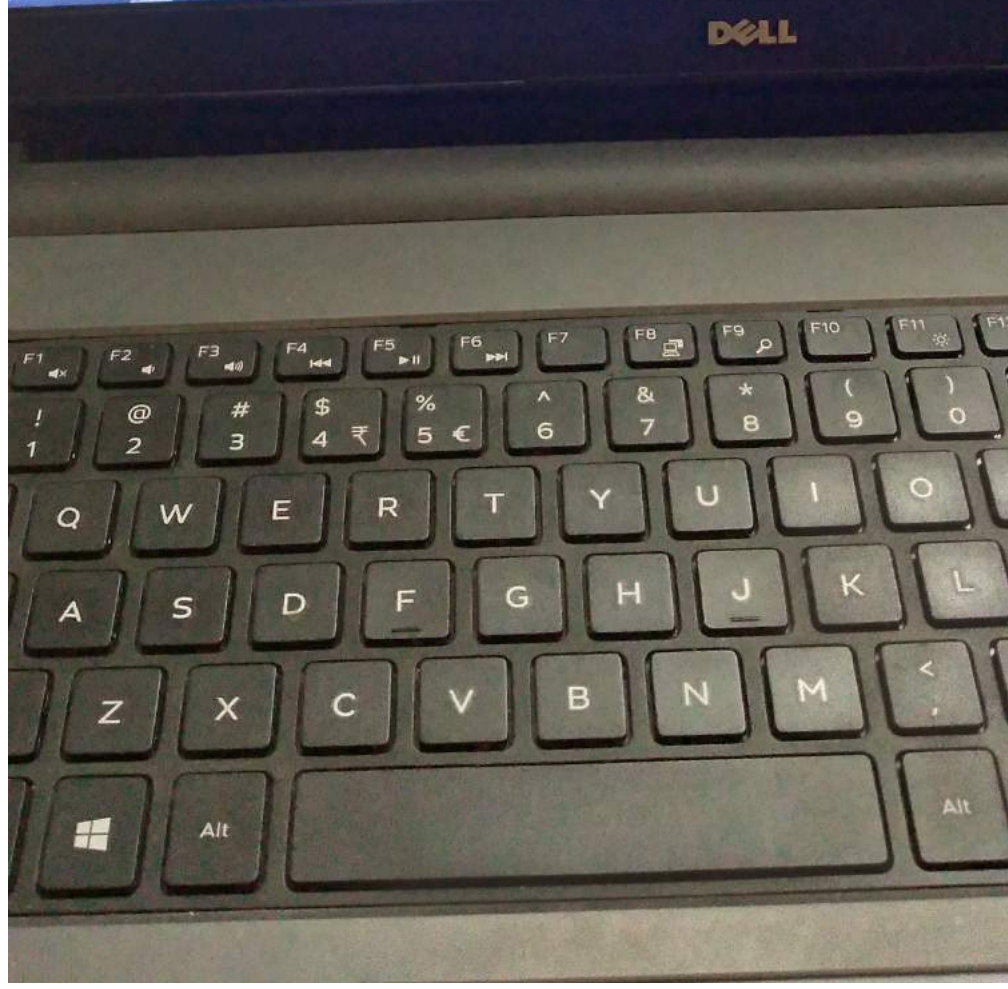
Cut Copy path Paste shortcut Move to Copy to Delete Rename New folder New Item Easy access Properties Open Select all Select none Invert selection

board

This PC > OS (C:) > Users > Ravi Raji

Name	Date modified	Type	Size
.oracle_jre_usage	12-10-2021 15:40	File folder	
.VirtualBox	14-02-2022 19:39	File folder	
.Xilinx	10-10-2021 02:34	File folder	
3D Objects	27-04-2021 14:03	File folder	
CHAITANYASUNDARJOSE	15-02-2022 09:28	File folder	
contacts	14-02-2022 19:39	File folder	
Contacts	27-04-2021 14:03	File folder	
Documents	09-10-2021 00:41	File folder	
Downloads	15-02-2022 09:26	File folder	
Dropbox	23-10-2016 19:42	File folder	
Favorites	27-04-2021 14:03	File folder	
Links	27-04-2021 14:03	File folder	
Music	29-07-2021 09:26	File folder	
New folder	07-01-2018 11:59	File folder	
New folder (2)	31-10-2021 11:50	File folder	
OneDrive	09-10-2021 00:40	File folder	
raji	21-08-2020 15:07	File folder	
Roaming	06-09-2016 21:18	File folder	
Saved Games	27-04-2021 14:03	File folder	
Searches	27-04-2021 14:03	File folder	
sudhakar	21-08-2020 14:27	File folder	
Videos	27-04-2021 14:03	File folder	
VirtualBox VMs	13-01-2022 15:06	File folder	
.bash_history	14-02-2022 19:34	BASH_HISTORY File	1 KB

1 item selected



MINGW64/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1

fsck

Ravi Raji@DESKTOP-9D4KJHO MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)

```
$ git config --list --show-
error: ambiguous option: show- (could be --show-origin or --show-scope)
usage: git config [<options>]
```

Config file location

--global	use global config file
--system	use system config file
--local	use repository config file
--worktree	use per-worktree config file
-f, --file <file>	use given config file
--blob <blob-id>	read config from given blob object

Action

--get	get value: name [value-pattern]
--get-all	get all values: key [value-pattern]
--get-regexp	get values for regexp: name-regex [value-pattern]
--get-urlmatch	get value specific for the URL: section[.var] URL
--replace-all	replace all matching variables: name value [value-pattern]
--add	add a new variable: name value
--unset	remove a variable: name [value-pattern]
--unset-all	remove all matches: name [value-pattern]
--rename-section	rename section: old-name new-name
--remove-section	remove a section: name
-l, --list	list all
--fixed-value	use string equality when comparing values to 'value-pattern'
-e, --edit	open an editor
--get-color	find the color configured: slot [default]
--get-colorbool	find the color setting: slot [stdout-is-atty]

Type

-t, --type <>	value is given this type
--bool	value is "true" or "false"
--int	value is decimal number
--bool-or-int	value is --bool or --int
--bool-or-str	value is --bool or string
--path	value is a path (file or directory name)
--expiry-date	value is an expiry date

Other

-z, --null	terminate values with NUL byte
--name-only	show variable names only
--includes	respect include directives on lookup
--show-origin	show origin of config (file, standard input, blob, command line)
--show-scope	show scope of config (worktree, local, global, system, command)
--default <value>	with --get, use default value when missing entry

MINGW64/c/Users/Ravi Raju/CHAITANYASUNDARJOSE/cognizance/task1

reference repository
--dissociate use --reference only while cloning
-o, --origin <name> use <name> instead of 'origin' to track upstream
-b, --branch <branch> checkout <branch> instead of the remote's HEAD
-u, --upload-pack <path> path to git-upload-pack on the remote
--depth <depth> create a shallow clone of that depth
--shallow-since <time> create a shallow clone since a specific time
--shallow-exclude <revision> deepen history of shallow clone, excluding rev
--single-branch clone only one branch, HEAD or --branch
--no-tags don't clone any tags, and make later fetches not to follow them
--shallow-submodules any cloned submodules will be shallow
--separate-git-dir <gitdir> separate git dir from working tree
-c, --config <key=value> set config inside the new repository
--server-option <server-specific> option to transmit
-4, --ipv4 use IPv4 addresses only
-6, --ipv6 use IPv6 addresses only
--filter <args> object filtering
--remote-submodules any cloned submodules will use their remote-tracking branch
--sparse initialize sparse-checkout file to include only files at root

Ravi Raju@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git fetch

Ravi Raju@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git pull
Already up to date.

Ravi Raju@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git merge
Already up to date.

Ravi Raju@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git rebase
Current branch master is up to date.

Ravi Raju@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git fork
git: 'fork' is not a git command. See 'git --help'.
The most similar command is
fsck

MINGW64: c:/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)

```
$ git remote -v
origin https://github.com/CHAITANYASUNDARJOSE/cognizance.git (fetch)
origin https://github.com/CHAITANYASUNDARJOSE/cognizance.git (push)
```

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)

```
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 264 bytes | 264.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/CHAITANYASUNDARJOSE/cognizance.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)

```
$ git checkout
Your branch is up to date with 'origin/master'.
```

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)

```
$ git clone
fatal: You must specify a repository to clone.
```

usage: git clone [<options>] [--] <repo> [<dir>]

-v, --verbose	be more verbose
-q, --quiet	be more quiet
--progress	force progress reporting
--reject-shallow	don't clone shallow repository
-n, --no-checkout	don't create a checkout
--bare	create a bare repository
--mirror	create a mirror repository (implies bare)
-l, --local	to clone from a local repository
--no-hardlinks	don't use local hardlinks, always copy
-s, --shared	setup as shared repository
--recurse-submodules[=<pathspec>]	initialize submodules in the clone
--recursive ...	alias of --recurse-submodules
-j, --jobs <n>	number of submodules cloned in parallel
--template <template-directory>	directory from which templates will be used
--reference <repo>	reference repository
--reference-if-able <repo>	reference repository
--dissociate	use --reference only while cloning
-o, --origin <name>	use <name> instead of 'origin' to track upstream
-b, --branch <branch>	checkout <branch> instead of the remote's HEAD
-u, --upload-pack <path>	

MINGW64/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1

Untracked files:

(use "git add <file>..." to include in what will be committed)
alpha.txt

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

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git add .

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git status
On branch master

No commits yet

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

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git commit -m "cognizance task1"
[master (root-commit) 99dee31] cognizance task1
1 file changed, 1 insertion(+)
create mode 100644 alpha.txt

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git log
commit 99dee317581e745af341a9a97bdc640ffce8837a (HEAD -> master)
Author: CHAITANYASUNDARJOSE <ch.en.u4cys21095@ch.students.amrita.edu>
Date: Tue Feb 15 09:33:07 2022 +0530

cognizance task1

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ touch task1.txt

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ notepad task1.txt

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ https://github.com/CHAITANYASUNDARJOSE/cognizance.git
bash: https://github.com/CHAITANYASUNDARJOSE/cognizance.git: No such file or directory

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git remote -v

Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
\$ git remote add origin https://github.com/CHAITANYASUNDARJOSE/cognizance.git


```
MINGW64:/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
$ pwd
/c/Users/Ravi Raji
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
$ mkdir CHAITANYASUNDARJOSE/cognizance
mkdir: cannot create directory 'CHAITANYASUNDARJOSE/cognizance': No such file or directory
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
$ mkdir CHAITANYASUNDARJOSE
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
$ cd CHAITANYASUNDARJOSE
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE (develop)
$ mkdir cognizance
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE (develop)
$ cd cognizance
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance (develop)
$ pwd
/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance (develop)
$ mkdir task1
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance (develop)
$ cd task1
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (develop)
$ pwd
/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (develop)
$ git init
Initialized empty Git repository in C:/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1/.git/
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
$ touch alpha.txt
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
$ notepad alpha.txt
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~/CHAITANYASUNDARJOSE/cognizance/task1 (master)
$ git status
On branch master
No commits yet
```



MINGW64/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1

```
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
git config --system user.name "CHAITANYASUNDARJOSE"
Error: could not lock config file C:/Program Files/Git/etc/gitconfig: Permission denied
```

```
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
git config --global user.name "CHAITANYASUNDARJOSE"
```

```
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
git config --global user.email ch.en.u4cys21095@ch.students.amrita.edu
Error: 'config' is not a git command. See 'git --help'.
```

The most similar command is
config

```
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
$ git config --global user.email ch.en.u4cys21095@ch.students.amrita.edu
```

```
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
user.name=CHAITANYASUNDARJOSE
user.email=ch.en.u4cys21095@ch.students.amrita.edu
gui.recentrepo=E:/amrita/OneDrive - Amrita Vishwa Vidyapeetham- Chennai Campus
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
```

```
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~ (develop)
```


MINGW64/c/Users/Ravi Raji/CHAITANYASUNDARJOSE/cognizance/task1

Ravi Raji@DESKTOP-9D4KJHD MINGW64 ~ (develop)

\$ git config --global user.name "CHAITANYASUNDARJOSE"

error: unknown option 'global'

usage: git config [<options>]

Config file location

--global	use global config file
--system	use system config file
--local	use repository config file
--worktree	use per-worktree config file
-f, --file <file>	use given config file
--blob <blob-id>	read config from given blob object

Action

--get	get value: name [value-pattern]
--get-all	get all values: key [value-pattern]
--get-regexp	get values for regexp: name-regex [value-pattern]
--get-urlmatch	get value specific for the URL: section[.var] URL
--replace-all	replace all matching variables: name value [value-pattern]
--add	add a new variable: name value
--unset	remove a variable: name [value-pattern]
--unset-all	remove all matches: name [value-pattern]
--rename-section	rename section: old-name new-name
--remove-section	remove a section: name
-l, --list	list all
--fixed-value	use string equality when comparing values to 'value-pattern'
-e, --edit	open an editor
--get-color	find the color configured: slot [default]
--get-colorbool	find the color setting: slot [stdout-is-tty]

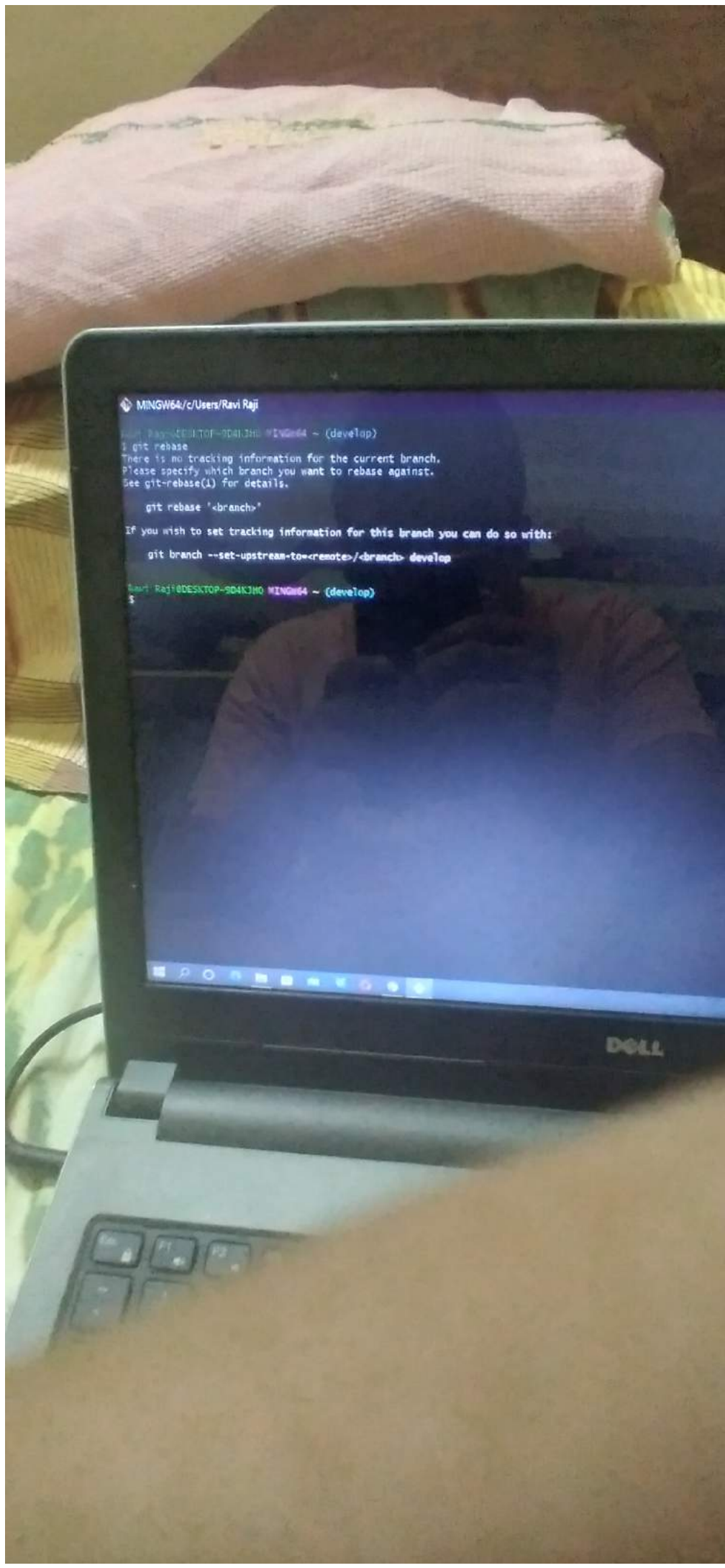
Type

-t, --type <>	value is given this type
--bool	value is "true" or "false"
--int	value is decimal number
--bool-or-int	value is --bool or --int
--bool-or-str	value is --bool or string
--path	value is a path (file or directory name)
--expiry-date	value is an expiry date

Other

-z, --null	terminate values with NUL byte
--name-only	show variable names only
--includes	respect include directives on lookup
--show-origin	show origin of config (file, standard input, blob, command line)
--show-scope	show scope of config (worktree, local, global, system, command)
--default <value>	with --get, use default value when missing entry

Ravi Raji@DESKTOP-9D4KJHD MINGW64 ~ (develop)



MINGW64/c/Users/Ravi Raji

~\$ git rebase

There is no tracking information for the current branch.
Please specify which branch you want to rebase against.
See git-rebase(1) for details.

git rebase '<branch>'

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=<remote>/<branch> develop

Ravi@DESKTOP-3D4KJH0 MINGW64 ~ (develop)

\$

MINGW64/c/Users/Ravi Raji

```
--reference <remote>      reference repository
--reference <remote> use --reference only while cloning
--origin <name>          use --name instead of 'origin' to track upstream
--branch <branch>        checkout <branch> instead of the remote's HEAD
--upload-pack <path>     path to git-upload-pack on the remote
--depth <depth>          create a shallow clone of that depth
--shallow-since <time>   create a shallow clone since a specific time
--shallow-exclude <revision>
                        deepen history of shallow clone, excluding rev
--single-branch           clone only one branch, HEAD or --branch
--no-tags                don't clone any tags, and make later fetches not to follow them
--shallow-submodules     any cloned submodules will be shallow
--separate-git-dir <gitdir>
                        separate git dir from working tree
-c, --config <key=value>
                        set config inside the new repository
--server-option <server-specific>
                        option to transmit
-4, --ipv4              use IPv4 addresses only
-6, --ipv6              use IPv6 addresses only
--filter <args>         object filtering
--remote-submodules     any cloned submodules will use their remote-tracking branch
--sparse                initialize sparse-checkout file to include only files at root
```

Ravi Raji@DESKTOP-904K3HO MINGW64 ~ (develop)

\$ git fetch

Ravi Raji@DESKTOP-904K3HO MINGW64 ~ (develop)

\$ git merge

fatal: No remote for the current branch.

Ravi Raji@DESKTOP-904K3HO MINGW64 ~ (develop)

\$ git pull

There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=<remote>/<branch> develop

Ravi Raji@DESKTOP-904K3HO MINGW64 ~ (develop)

\$

MINGW64/c/Users/Ravi Raji

```
git checkout -b develop  
Switched to a new branch 'develop'
```

```
git push origin develop  
error: src refspec develop does not match any
```

```
git clone  
fatal: You must specify a repository to clone.
```

usage: git clone [<options>] [--] <repo> [<dir>]

--v, --verbose	be more verbose
-q, --quiet	be more quiet
--progress	force progress reporting
--reject-shallow	don't clone shallow repository
-n, --no-checkout	don't create a checkout
--bare	create a bare repository
--mirror	create a mirror repository (implies bare)
-l, --local	to clone from a local repository
--no-hardlinks	don't use local hardlinks, always copy
-s, --shared	setup as shared repository
--recurse-submodules[=<pathspec>]	initialize submodules in the clone
--recursive ...	alias of --recurse-submodules
-j, --jobs <n>	number of submodules cloned in parallel
--template <template-directory>	directory from which templates will be used
--reference <repo>	reference repository
--reference-if-able <repo>	reference repository
--dissociate	use --reference only while cloning
-o, --origin <name>	use <name> instead of 'origin' to track upstream
-b, --branch <branch>	checkout <branch> instead of the remote's HEAD
-u, --upload-pack <path>	path to git-upload-pack on the remote
--depth <depth>	create a shallow clone of that depth
--shallow-since <time>	create a shallow clone since a specific time
--shallow-exclude <revision>	deepen history of shallow clone, excluding rev
--single-branch	clone only one branch, HEAD or --branch
--no-tags	don't clone any tags, and make later fetches not to follow them
--shallow-submodules	any cloned submodules will be shallow
--separate-git-dir <gitdir>	separate git dir from working tree

MINGW64/c/Users/Ravi Raji

```
warning: could not open directory 'Local Settings/': Permission denied
warning: could not open directory 'My Documents/': Permission denied
warning: could not open directory 'NetHood/': Permission denied
warning: could not open directory 'PrintHood/': Permission denied
warning: could not open directory 'Recent/': Permission denied
warning: could not open directory 'SendTo/': Permission denied
warning: could not open directory 'Start Menu/': Permission denied
warning: could not open directory 'Templates/': Permission denied
On branch master
```

Initial commit

Untracked files:

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

```
.VirtualBox/
.vbox/
.vbox_history
.gitconfig
oracle_jre_usage/
10TH ANNEXURE 1&2.xlsx
3D Objects/
AppData/
Contacts/
Downloads/
Favorites/
IntelGraphicsProfiles/
Links/
MicrosoftEdgeBackups/
Music/
NTUSER.DAT
NTUSER.DAT{8a655988-a72e-11eb-8b30-b0f7d40e529f}.TM.b1f
NTUSER.DAT{8a655988-a72e-11eb-8b30-b0f7d40e529f}.TMContainer000000000000000001.regtrans-ms
NTUSER.DAT{8a655988-a72e-11eb-8b30-b0f7d40e529f}.TMContainer000000000000000002.regtrans-ms
New Folder/
Searches/
git_Trace.log
Videos/
VirtualBox VMs/
VirtualBox-6.1.26-167626-win.exe
ntuser.dat.LOG1
ntuser.dat.LOG2
ntuser.txt
reg/
subtree/
```

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

Ravi Raji@DESKTOP-3D4CH19: MINGW64 - (master)

5


```
MINGW64/c/Users/Ravi Raj:/
C:\Users\Ravi Raj> git init
Initialized empty Git repository in C:\Users\Ravi Raj\.git\

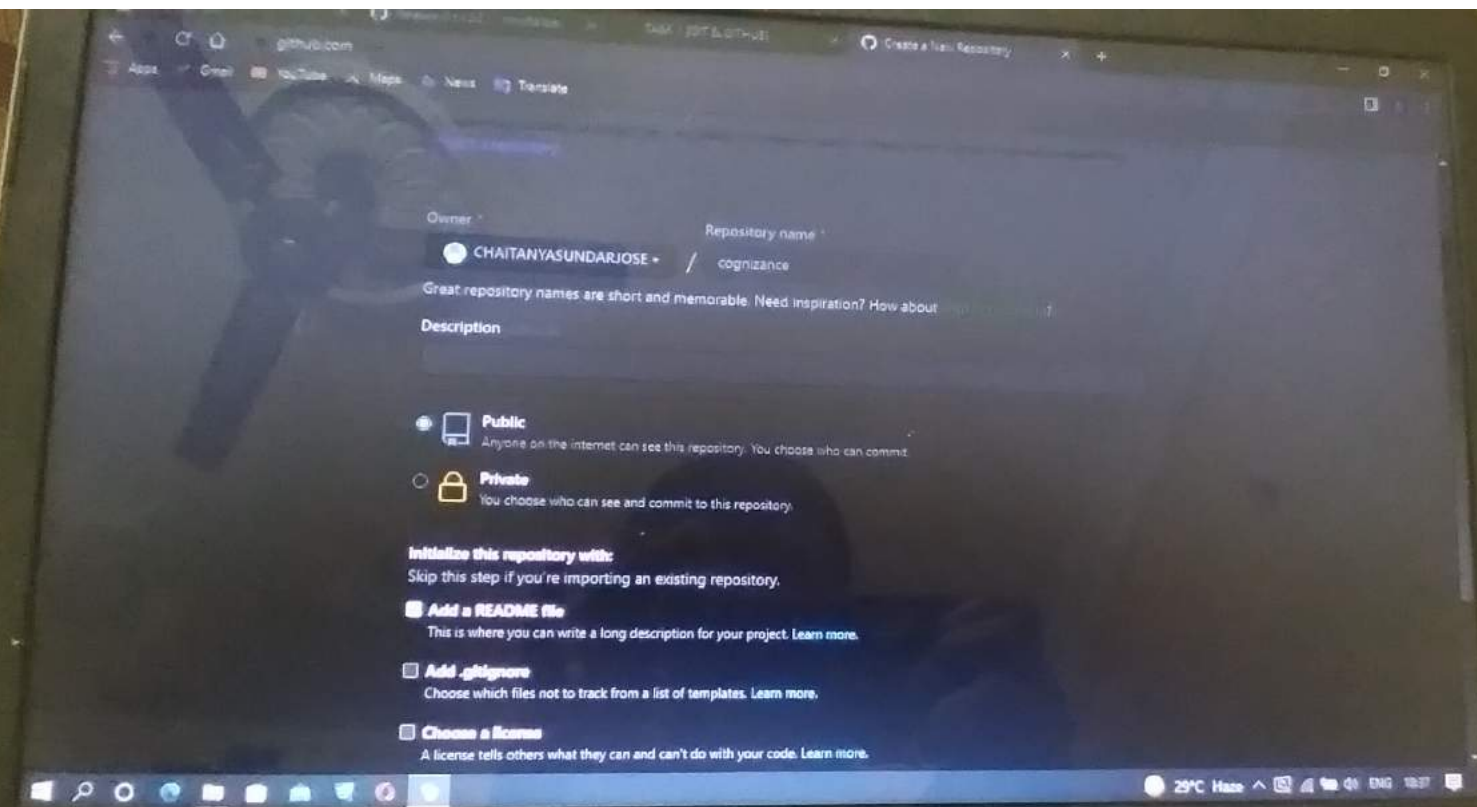
C:\Users\Ravi Raj> cd C:\ProgramData\Foxit Software\Foxit Reader
C:\ProgramData\Foxit Software\Foxit Reader> git add myFile_2.txt
fatal: pathspec 'myFile_2.txt' did not match any files

C:\ProgramData\Foxit Software\Foxit Reader> git status
warning: could not open directory 'AppData/Local/Application Data/': Permission denied
warning: could not open directory 'Application Data/': Permission denied
warning: could not open directory 'Cookies/': Permission denied
warning: could not open directory 'Documents/My Music/': Permission denied
warning: could not open directory 'Documents/My Pictures/': Permission denied
warning: could not open directory 'Documents/My Videos/': Permission denied
warning: could not open directory 'Local Settings/': Permission denied
warning: could not open directory 'My Documents/': Permission denied
warning: could not open directory 'NetHood/': Permission denied
warning: could not open directory 'PrintHood/': Permission denied
warning: could not open directory 'Recent/': Permission denied
warning: could not open directory 'SendTo/': Permission denied
warning: could not open directory 'Start Menu/': Permission denied
warning: could not open directory 'Templates/': Permission denied
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    VirtualBox/
    .Xslim/
    .bash_history
    .gitconfig
    .oracle_pre_image/
    10m ANCHORS 182.kloc
    3D Objects/
    Audata/
    Contacts/
    Downloads/
    Favorites/
    IntelliMouseProfiles/
    Links/
    MicrosoftEdgeBackup/
    Music/
    MYFILES.BAT
    MYFILES.DAT
    MYFILES.DIR [e7d1c88-e72c-11e6-b9db-bf7aaab2cf]..Tn.k7d
    MYFILES.DIR [a6c1018-e72c-11e6-b9db-bf7aaab2cf]..Tn.cntainer-GamesAndApps-000002_registry-ms
    MYFILES.DIR [a6c1018-e72c-11e6-b9db-bf7aaab2cf]..Tn.cntainer-GamesAndApps-000002_registry-ms
    New Folder
```

```
MINGW64/c/Users/Ravi Raji
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~
$ pwd
/c/Users/Ravi Raji
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~
$ mkdir cognizance
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~
$ ls
10TH ANNEXURE 1A2.xlsx  TO Objects/  Application Data/  Contacts/  Cookies/  Documents/  Downloads/  Dropbox/  Favorites/  IntelGraphicsProfiles/  Links/  Local Settings/  MicrosoftEdgeBackups/  Music/  My Documents/  NTUSER.DAT  NTUSER.DAT{8a655988-a72e-11eb-9b30-b077d40e529f}.TH.b7p  NTUSER.DAT{8a655988-a72e-11eb-9b30-b077d40e529f}.TH.Container00000000000000000001.regtrans-ms  NTUSER.DAT{8a655988-a72e-11eb-9b30-b077d40e529f}.TH.Container00000000000000000002.regtrans-ms  NetHood/  New folder/
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~
$ cd cognizance
bash: cd: cognizance: No such file or directory
Ravi Raji@DESKTOP-9D4KJH0 MINGW64 ~
$
```

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *



CHAITANYASUNDARJOSE ▾

Repository name *

cognizance1



Great repository names are short and memorable. **cognizance1** is available. How about **bug-free-succotash**?

Description (optional)



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:

Git-2.35.1.2-64-bit....exe

Canceled