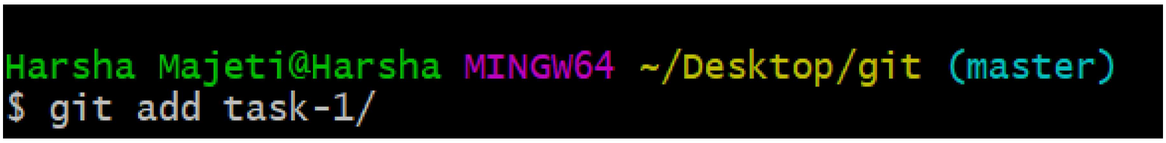
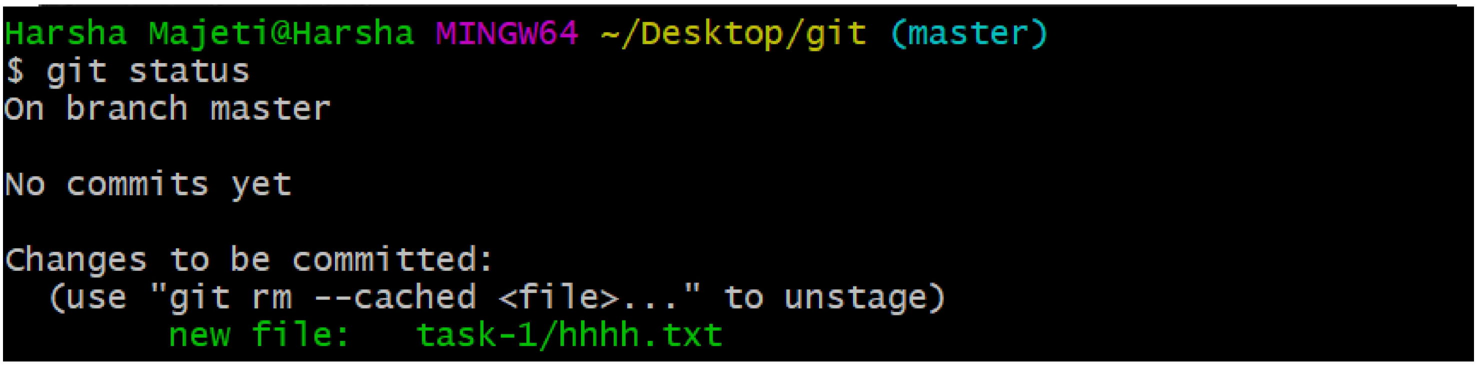
GIT Commands for Task 1

Git add



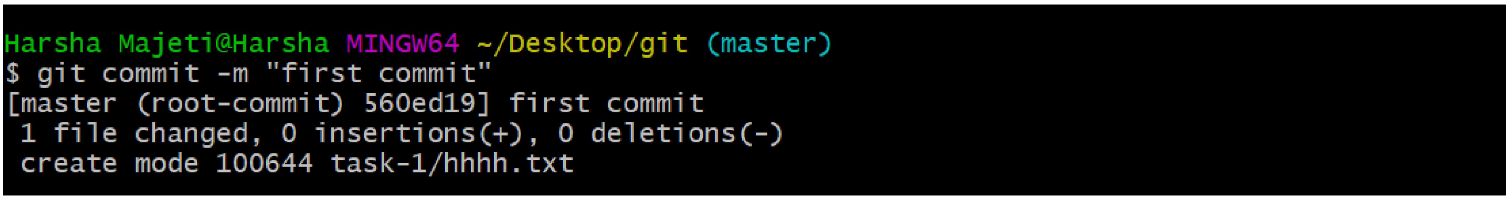
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git add task-1/
```

Git status



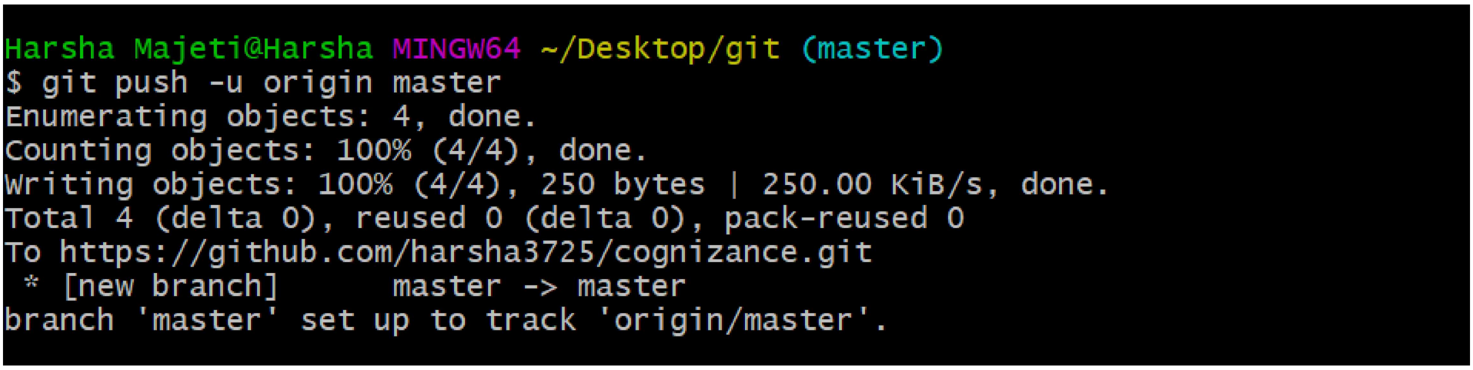
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   task-1/hhhh.txt
```

Git commit



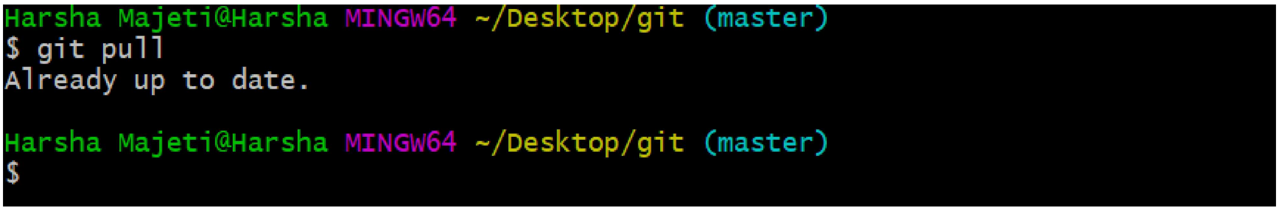
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git commit -m "first commit"
[master (root-commit) 560ed19] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 task-1/hhhh.txt
```

Git push



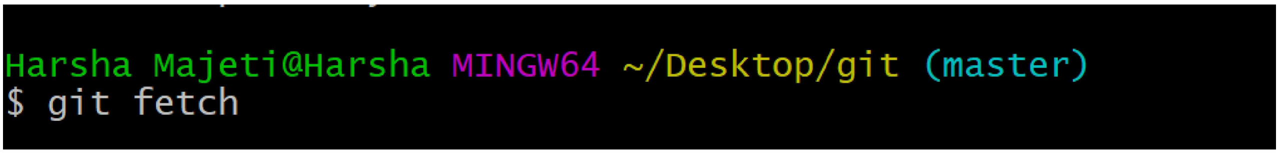
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git push -u origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (4/4), 250 bytes | 250.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/harsha3725/cognizance.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Git pull



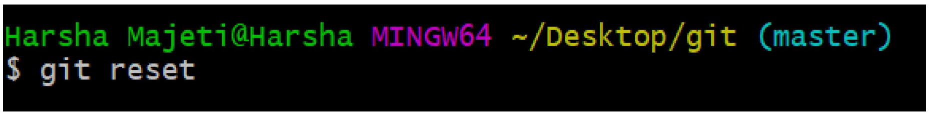
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git pull
Already up to date.

Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$
```

Git fetch



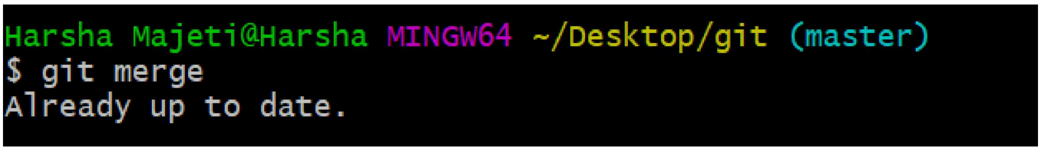
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git fetch
```

Git reset



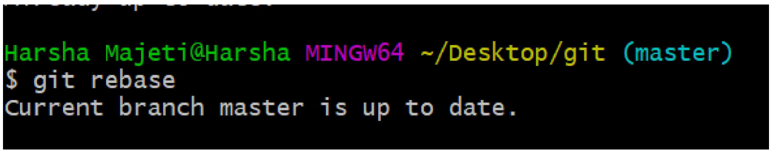
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git reset
```

Git merge



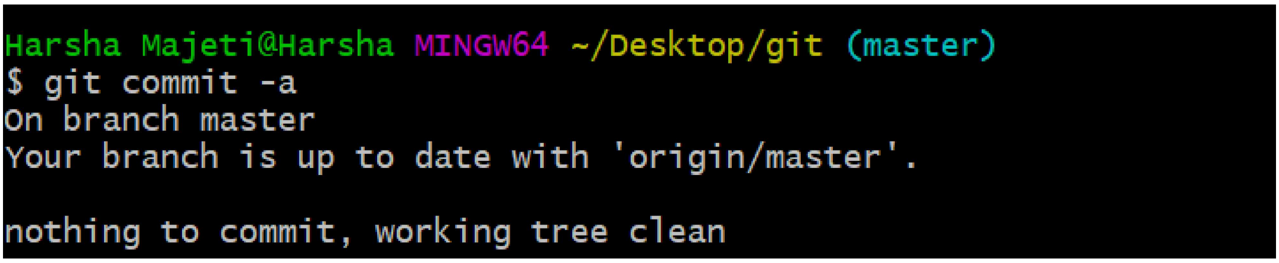
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git merge
Already up to date.
```

Git rebase



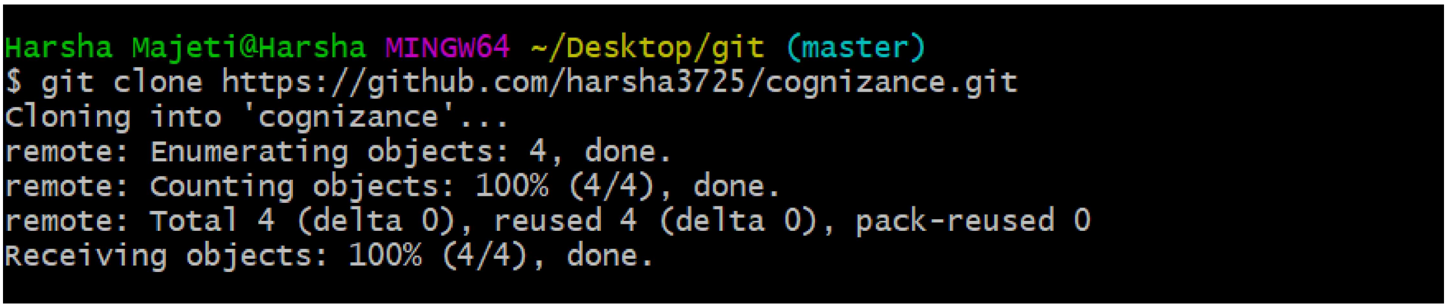
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git rebase
Current branch master is up to date.
```

Git commit -a



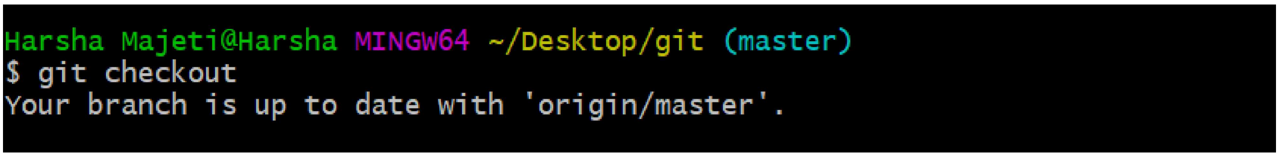
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git commit -a
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

Git clone



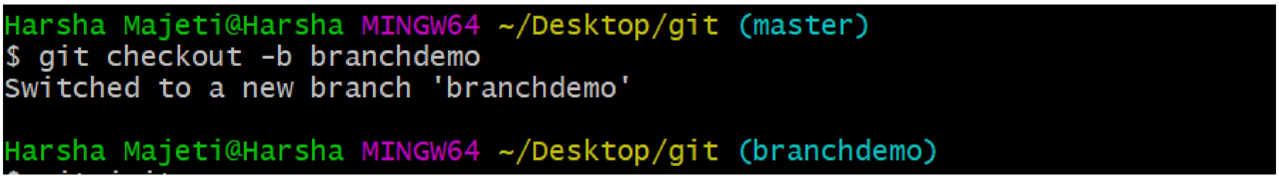
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git clone https://github.com/harsha3725/cognizance.git
Cloning into 'cognizance'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 4 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.
```

Git checkout



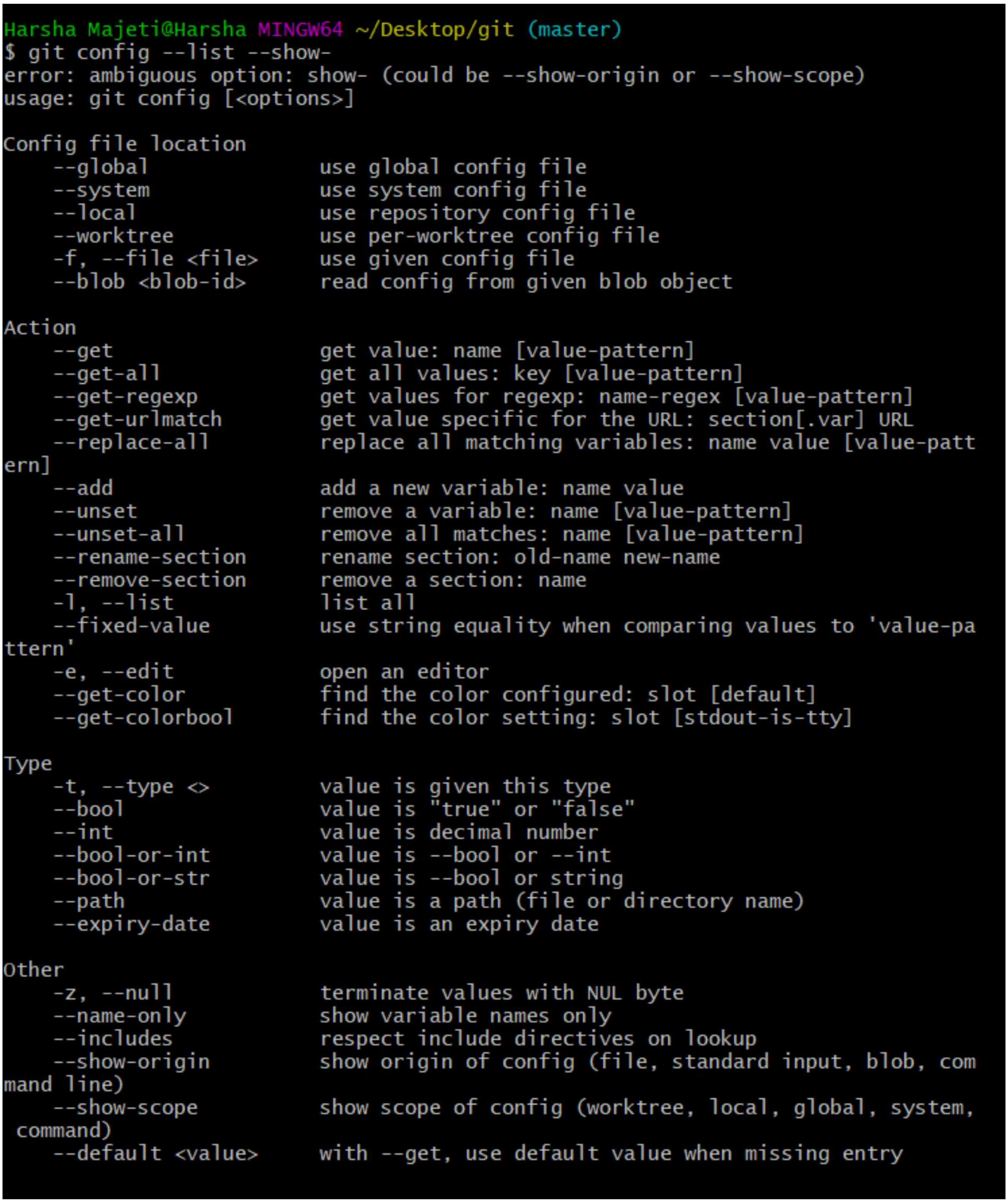
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git checkout
Your branch is up to date with 'origin/master'.
```

Git checkout -b



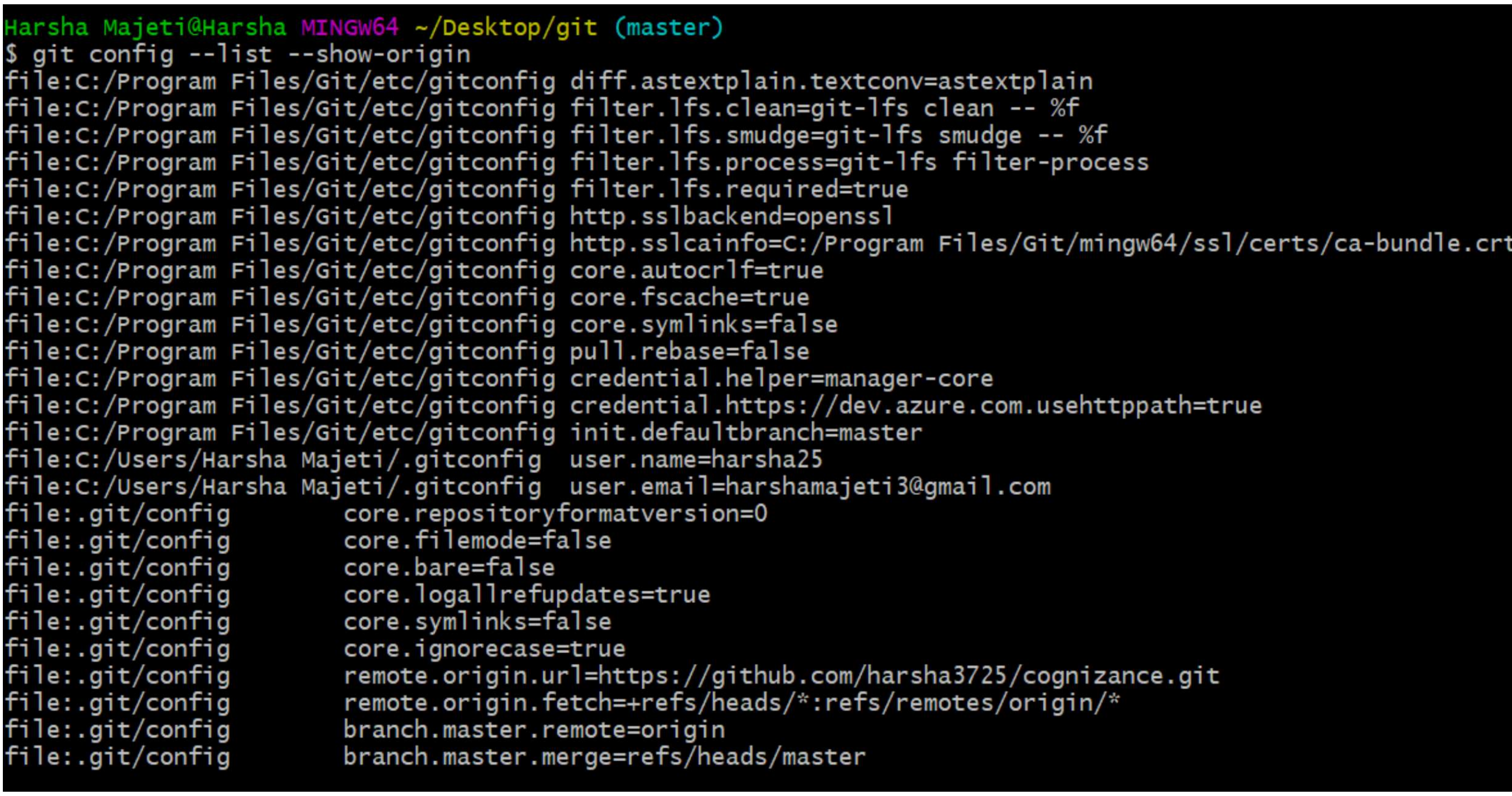
```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (master)
$ git checkout -b branchdemo
Switched to a new branch 'branchdemo'

Harsha Majeti@Harsha MINGW64 ~/Desktop/git (branchdemo)
```

git config --list --show-



```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (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-patt
ern]
    --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-pa
ttern'
    -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, com
mand line)
    --show-scope          show scope of config (worktree, local, global, system,
 command)
    --default <value>     with --get, use default value when missing entry
```

git config --list --show-origin



```
Harsha Majeti@Harsha MINGW64 ~/Desktop/git (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/Harsha Majeti/.gitconfig  user.name=harsha25
file:C:/Users/Harsha Majeti/.gitconfig  user.email=harshamajeti3@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/harsha3725/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
```