

GIT INIT:

```
MINGW64:/c/Users/ragul/cognizance

ragul@RagulMuthusami MINGW64 ~
$ pwd
/c/Users/ragul

ragul@RagulMuthusami MINGW64 ~
$ mkdir cognizance
mkdir: cannot create directory 'cognizance': File exists

ragul@RagulMuthusami MINGW64 ~
$ ls
'3D Objects'/'My Documents'@
'AppData'/'NTUSER.DAT'
'Application Data'@ NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TxR.0.regtrans-ms
'Contacts'/'NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TxR.1.regtrans-ms
'Cookies'@ NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TxR.2.regtrans-ms
'Documents'/'NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TxR.b1f
'Downloads'/'NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TM.b1f
'Dropbox'/'NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TMContainer000000000000000001.regtrans-ms
'Favorites'/'NTUSER.DAT{1c2b59c5-c5f5-11eb-bacb-000d3a96488e}.TMContainer000000000000000002.regtrans-ms
'IntelGraphicsProfiles'/'NetHood@
'Links'/'OneDrive/'
'Local Settings'@ 'PrintHood@
'Music/'Recent@
'Saved Games'/'
'Searches'/'
'SendTo@
'Start Menu'@
'Templates@
'Videos'/'
'VirtualBox VMs'/'
cognizance/
ntuser.dat.LOG1
ntuser.dat.LOG2
ntuser.ini

ragul@RagulMuthusami MINGW64 ~
$ cd cognizance

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

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$
```

ADD:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ touch test.txt


ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git add.
git: 'add.' is not a git command. See 'git --help'.

The most similar command is
    add

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

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$
```

GIT STATUS AND USERNAME AND USER EMAIL CONFIGURE:

 MINGW64:/c/Users/ragul/cognizance

```
ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ ls
test.txt

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

No commits yet

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

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git log
fatal: your current branch 'master' does not have any commits yet

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git config --global user.name ragulmuthusami

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git config --global user.email ragulmagesh2003@gmail.com

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

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git commit -a
[master (root-commit) 6295112] this is a text file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 test.txt

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$
```

CONNECTING LOCAL REPOSITORY TO REMOTE REPOSITORY:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ echo "# cognizance" >> README.md

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git init
Reinitialized existing Git repository in C:/Users/ragul/cognizance/.git/

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git add README.md
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git commit -m "first commit"
[master 0b7060f] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md

ragul@RagulMuthusami MINGW64 ~/cognizance (master)
$ git branch -M main

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git remote add origin https://github.com/ragulmuthusami/cognizance.git

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git push -u origin main
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (6/6), 476 bytes | 238.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/ragulmuthusami/cognizance.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$
```

The screenshot shows a web browser displaying the GitHub repository page for 'ragulmuthusami/cognizance'. The repository is in the 'main' branch, has 1 branch, and 0 tags. The commit history shows a single commit 'ragulmuthusami first commit' (0b7060f) from 6 minutes ago. The commit details show two files: 'README.md' (first commit, 6 minutes ago) and 'test.txt' (this is a text file, 18 minutes ago). The README.md content is visible, showing the repository name 'cognizance'. The right sidebar contains the 'About' section (No description, website, or topics provided), 'Releases' (No releases published), and 'Packages' (No packages published).

CHECKOUT:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git log --oneline
0b7060f (HEAD -> main, origin/main) first commit
6295112 this is a text file

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git checkout 6295112
Note: switching to '6295112'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 6295112 this is a text file

ragul@RagulMuthusami MINGW64 ~/cognizance ((6295112...))
$
```

CHECKOUT B:

COMMIT A:

MERGE:

```
ragul@RagulMuthusami MINGW64 ~/cognizance ((6295112...))
$ git checkout -b f1
Switched to a new branch 'f1'

ragul@RagulMuthusami MINGW64 ~/cognizance (f1)
$ touch test2.txt

ragul@RagulMuthusami MINGW64 ~/cognizance (f1)
$ git add .

ragul@RagulMuthusami MINGW64 ~/cognizance (f1)
$ git commit -a
[f1 8a366bd] this is a test2 file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 test2.txt

ragul@RagulMuthusami MINGW64 ~/cognizance (f1)
$ git branch -a
* f1
  main
remotes/origin/main

ragul@RagulMuthusami MINGW64 ~/cognizance (f1)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git merge f1
Merge made by the 'ort' strategy.
test2.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 test2.txt

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ A|
```

PUSH:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git push
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 526 bytes | 526.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/ragulmuthusami/cognizance.git
0b7060f..7ca32ae main -> main


ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$
```


PULL:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git pull
Already up to date.

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ |
```

GIT CONFIG --LIST --SHOW-ORIGIN:

 MINGW64:/c/Users/ragul/cognizance

```
ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git config --list --show-origin
error: unknown non-ascii option in string: '--show-origin'
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
```

CLONE:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git clone https://github.com/ragulmuthusami/cognizance.git
Cloning into 'cognizance'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 10 (delta 3), reused 7 (delta 0), pack-reused 0
Receiving objects: 100% (10/10), done.
Resolving deltas: 100% (3/3), done.

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$
```

RESET:

```
ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git log --oneline
7ca32ae (HEAD -> main, origin/main) Merge branch 'f1'
8a366bd (f1) this is a test2 file
0b7060f first commit
6295112 this is a text file

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$ git reset 8a366bd --hard
HEAD is now at 8a366bd this is a test2 file

ragul@RagulMuthusami MINGW64 ~/cognizance (main)
$
```