

Task 1

```
prans@DESKTOP-KD3PIPF MINGW64 ~ (master)
$ mkdir cognizance

prans@DESKTOP-KD3PIPF MINGW64 ~ (master)
$ cd /cognizance
bash: cd: /Cognizance: No such file or directory

prans@DESKTOP-KD3PIPF MINGW64 ~ (master)
$ cd Cognizance/

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git init
Initialized empty Git repository in c:/Users/prans/cognizance/.git/

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ touch welcome.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git commit -m "new commit"
on branch master

Initial commit

untracked files:
  (use "git add <file>..." to include in what will be committed)
    welcome.txt

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

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git remote add origin git@github.com:Beawareofme/cognizance.git

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ curl -u Beawareofme:Pranshulatri@123 https://api.github.com/user/repos -d '{"name":"cognizance"}'
  % Total    % Received % Xferd  Average Speed   Time    Time     Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 182 100 161 100 21 342 44 --:--:-- --:--:-- --:--:-- 388{
  "message": "Requires authentication",
  "documentation_url": "https://docs.github.com/rest/reference/repos#create-a-repository-for-the-authenticated-user"
}
```

```
prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ curl -u Beawareofme:ghp_cpMxxb1PvS8BwS1A2nsuHYJmbxrv0nPdVz https://api.github.com/user/repos -d '{"name":"cognizance"}'
  % Total    % Received % Xferd  Average Speed   Time    Time     Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 21 0 0 100 21 0 17 0:00:01 0:00:01 --:--:-- 17{
  "id": 45949117,
  "node_id": "R_kgDOG2NenQ",
  "name": "Cognizance",
  "full_name": "Beawareofme/cognizance",
  "private": false,
  "owner": {
    "login": "Beawareofme",
    "id": 99644054,
    "node_id": "U_kg008f8yIg",
    "avatar_url": "https://avatars.githubusercontent.com/u/99644054?v=4",
    "gravatar_id": "",
    "url": "https://api.github.com/users/Beawareofme",
    "html_url": "https://github.com/Beawareofme",
    "followers_url": "https://api.github.com/users/Beawareofme/followers",
    "following_url": "https://api.github.com/users/Beawareofme/following{/other_user}",
    "gists_url": "https://api.github.com/users/Beawareofme/gists{/gist_id}",
    "starred_url": "https://api.github.com/users/Beawareofme/starred{/owner}/{/repo}",
    "subscriptions_url": "https://api.github.com/users/Beawareofme/subscriptions",
    "organizations_url": "https://api.github.com/users/Beawareofme/orgs",
    "repos_url": "https://api.github.com/users/Beawareofme/repos",
    "events_url": "https://api.github.com/users/Beawareofme/events{/privacy}",
    "received_events_url": "https://api.github.com/users/Beawareofme/received_events",
    "type": "User",
    "site_admin": false
  },
  "html_url": "https://github.com/Beawareofme/cognizance",
  "description": null,
  "fork": false,
  "url": "https://api.github.com/repos/Beawareofme/cognizance",
  "forks_url": "https://api.github.com/repos/Beawareofme/cognizance/forks",
  "keys_url": "https://api.github.com/repos/Beawareofme/cognizance/keys{/key_id}",
  "collaborators_url": "https://api.github.com/repos/Beawareofme/cognizance/collaborators{/collaborator}",
  "teams_url": "https://api.github.com/repos/Beawareofme/cognizance/teams",
  "hooks_url": "https://api.github.com/repos/Beawareofme/cognizance/hooks",
  "issue_events_url": "https://api.github.com/repos/Beawareofme/cognizance/issues/events{/number}",
```

```

"issue_events_url": "https://api.github.com/repos/Beawareofme/Cognizance/issues/events{/number}",
"events_url": "https://api.github.com/repos/Beawareofme/Cognizance/events",
"assignees_url": "https://api.github.com/repos/Beawareofme/Cognizance/assignees{/user}",
"branches_url": "https://api.github.com/repos/Beawareofme/Cognizance/branches{/branch}",
"tags_url": "https://api.github.com/repos/Beawareofme/Cognizance/tags",
"blobs_url": "https://api.github.com/repos/Beawareofme/Cognizance/git/blobs{/sha}",
"git_tags_url": "https://api.github.com/repos/Beawareofme/Cognizance/git/tags{/sha}",
"git_refs_url": "https://api.github.com/repos/Beawareofme/Cognizance/git/refs{/sha}",
"trees_url": "https://api.github.com/repos/Beawareofme/Cognizance/git/trees{/sha}",
"statuses_url": "https://api.github.com/repos/Beawareofme/Cognizance/statuses{/sha}",
"languages_url": "https://api.github.com/repos/Beawareofme/Cognizance/languages",
"stargazers_url": "https://api.github.com/repos/Beawareofme/Cognizance/stargazers",
"contributors_url": "https://api.github.com/repos/Beawareofme/Cognizance/contributors",
"subscribers_url": "https://api.github.com/repos/Beawareofme/Cognizance/subscribers",
"subscription_url": "https://api.github.com/repos/Beawareofme/Cognizance/subscription",
"commits_url": "https://api.github.com/repos/Beawareofme/Cognizance/commits{/sha}",
"git_commits_url": "https://api.github.com/repos/Beawareofme/Cognizance/git/commits{/sha}",
"comments_url": "https://api.github.com/repos/Beawareofme/Cognizance/comments{/number}",
"issue_comment_url": "https://api.github.com/repos/Beawareofme/Cognizance/issues/comments{/number}",
"contents_url": "https://api.github.com/repos/Beawareofme/Cognizance/contents/{+path}",
"compare_url": "https://api.github.com/repos/Beawareofme/Cognizance/compare/{base}...{head}",
"merges_url": "https://api.github.com/repos/Beawareofme/Cognizance/merges",
"archive_url": "https://api.github.com/repos/Beawareofme/Cognizance/{archive_format}{ref}",
"downloads_url": "https://api.github.com/repos/Beawareofme/Cognizance/downloads",
"issues_url": "https://api.github.com/repos/Beawareofme/Cognizance/issues{/number}",
"pulls_url": "https://api.github.com/repos/Beawareofme/Cognizance/pulls{/number}",
100 5797 100 576 100 21 4296 15 0:00:01 0:00:01 --:--:-- 4316ps://api.github.com/repos/Beawareofme/Cognizance/milestones{/number}
"notifications_url": "https://api.github.com/repos/Beawareofme/Cognizance/notifications?since=all,participating",
"labels_url": "https://api.github.com/repos/Beawareofme/Cognizance/labels/{name}",
"releases_url": "https://api.github.com/repos/Beawareofme/Cognizance/releases/{id}",
"deployments_url": "https://api.github.com/repos/Beawareofme/Cognizance/deployments",
"created_at": "2022-02-15T08:46:26Z",
"updated_at": "2022-02-15T08:46:26Z",
"pushed_at": "2022-02-15T08:46:26Z",
"git_url": "git://github.com/Beawareofme/Cognizance.git",
"ssh_url": "git@github.com:Beawareofme/Cognizance.git",
"clone_url": "https://github.com/Beawareofme/Cognizance.git",
"svn_url": "https://github.com/Beawareofme/Cognizance",
"homepage": null,
"size": 0,

```

```

"stargazers_count": 0,
"watchers_count": 0,
"language": null,
"has_issues": true,
"has_projects": true,
"has_downloads": true,
"has_wiki": true,
"has_pages": false,
"forks_count": 0,
"mirror_url": null,
"archived": false,
"disabled": false,
"open_issues_count": 0,
"license": null,
"allow_forking": true,
"is_template": false,
"topics": [
],
"visibility": "public",
"forks": 0,
"open_issues": 0,
"watchers": 0,
"default_branch": "main",
"permissions": {
  "admin": true,
  "maintain": true,
  "push": true,
  "triage": true,
  "pull": true
},
"allow_squash_merge": true,
"allow_merge_commit": true,
"allow_rebase_merge": true,
"allow_auto_merge": false,
"delete_branch_on_merge": false,
"allow_update_branch": false,
"network_count": 0,
"subscribers_count": 0
}

```

```
prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$
{"has_issues": true,
"has_projects": true,
"has_downloads": true,
"has_wiki": true,
"has_pages": false,
"forks_count": 0,
"mirror_url": null,
"archived": false,
"disabled": false,
"open_issues_count": 0,
"license": null,
"allow_forking": true,
"is_template": false,
"topics": [
],
"visibility": "public",
"forks": 0,
"open_issues": 0,
"watchers": 0,
"default_branch": "main",
"permissions": {
  "admin": true,
  "maintain": true,
  "push": true,
  "triage": true,
  "pull": true
},
"allow_squash_merge": true,
"allow_merge_commit": true,
"allow_rebase_merge": true,
"allow_auto_merge": false,
"delete_branch_on_merge": false,
"allow_update_branch": false,
"network_count": 0,
"subscribers_count": 0
}
```

Task 2

```
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch Task1

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git init
Reinitialized existing Git repository in c:/Users/prans/cognizance/.git/

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add Task1

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git commit -m "First task commit"
[master (root-commit) f0c615e] First task commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Task1

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git remote add origin https://github.com/Beawareofme/Cognizance.git
error: remote origin already exists.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push -u origin master
The authenticity of host 'github.com (13.234.210.38)' can't be established.
ED25519 key fingerprint is SHA256:+D1Y3wvvv6Tuj3hbpZ1sF/zLDA0zPMSvHdkr4uvcoqu.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
git@github.com: Permission denied (publickey).
fatal: could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ssh-keygen -t ed25519 -C "pranshu19109193@gmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c:/Users/prans/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
Your identification has been saved in /c:/Users/prans/.ssh/id_ed25519
Your public key has been saved in /c:/Users/prans/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:kNBHwFguqdhxsm9rF6+iEw3MF5po9C/8Fooc2945Rk pranshu19109193@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
|      .oo'o      |
|      + =       |
|      + O O      |
|      O B = O O   |
|      + + S =     |
|      + * *E . .  |
|      O =.= O +   |
|      O.=O.=      |
|      O+.= O.     |
+----[SHA256]-----+

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ eval "$(ssh-agent -s)"
Agent pid 1382

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ssh-add ~/.ssh/id_ed25519
Identity added: /c:/Users/prans/.ssh/id_ed25519 (pranshu19109193@gmail.com)

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ cat ~/.ssh/id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAID05MN7CP0m3ROV4yykj8dvzvsVS5HohqIqsmMRJAUTs pranshu19109193@gmail.com

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 216 bytes | 216.00 kib/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Beawareofme/Cognizance.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push origin master

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push origin master
Everything up-to-date

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$
```

Task 3

```
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (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-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

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

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$
```

Task 4

Here I have used add command, commit command

Add command: to shift the file to be staged.

Commit command: Commits staged changes to a local branch.

```
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
bash: ls: command not found

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
Task1  welcome.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ cd Task1
bash: cd: Task1: Not a directory

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ cd Task1/
bash: cd: Task1/: Not a directory

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ cd /Task1
bash: cd: /Task1: No such file or directory

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch hello.py

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (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)
    welcome.txt
    bye.c
    hello.py
    index.html
```

```
nothing added to commit but untracked files present (use "git add" to track)
prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git add -A

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   welcome.txt
        new file:   bye.c
        new file:   hello.py
        new file:   index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   welcome.txt
        new file:   bye.c
        new file:   hello.py
        new file:   index.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   bye.c
        modified:   hello.py
        modified:   index.html
```

```
prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git add -A

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   welcome.txt
        new file:   bye.c
        new file:   hello.py
        new file:   index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git remote add origin https://github.com/Beawareofme/cognizance.git
error: remote origin already exists.

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git push -u origin master
Everything up-to-date
branch 'master' set up to track 'origin/master'.

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   welcome.txt
        new file:   bye.c
        new file:   hello.py
        new file:   index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git commit -a -m "Task1 part4 commit"
```

```
prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git commit -a -m "Task1 part4 commit"
[master ba0d8de] Task1 part4 commit
4 files changed, 28 insertions(+)
create mode 100644 welcome.txt
create mode 100644 bye.c
create mode 100644 hello.py
create mode 100644 index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git remote add origin https://github.com/Beawareofme/cognizance.git
error: remote origin already exists.

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$ git push -u origin master
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% (5/5), 664 bytes | 664.00 kib/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Beawareofme/cognizance.git
 f0c615e..ba0d8de  master -> master
branch 'master' set up to track 'origin/master'.

prans@DESKTOP-KD3PIPF MINGW64 ~/cognizance (master)
$
```

Push command usage: Uploads changes from all local branches to the respective remote repositories.

```

MINGW64/C:/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   bye.c

no changes added to commit (use "git add" and/or "git commit -a")

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git commit -a -m "Made some changes to bye.c file"
[master 3803cb0] Made some changes to bye.c file
1 file changed, 14 insertions(+), 15 deletions(-)

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 430 bytes | 430.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:Beawareofme/Cognizance.git
ba0d8de..3803cb0 master -> master

```

```

MINGW64/C:/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout bye.c
Updated 0 paths from the index

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   bye.c

no changes added to commit (use "git add" and/or "git commit -a")

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout bye.c
Updated 0 paths from the index

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)

```

```
changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   bye.c

changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   bye.c

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout bye.c
Updated 1 path from the index

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   bye.c

changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   bye.c
    modified:   hello.py
    modified:   index.html
```

```
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout --help

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git branch Test1

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git branch
  Test1
* master

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout Test1
Switched to branch 'Test1'
M    bye.c
M    hello.py
M    index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Test1)
$ git branch master
fatal: a branch named 'master' already exists

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Test1)
$ git checkout master
Switched to branch 'master'
M    bye.c
M    hello.py
M    index.html
Your branch is up to date with 'origin/master'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout -b Test1
fatal: a branch named 'Test1' already exists

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$
```

Here I have used pull command, fetch command and checkout command

Pull command: Fetches remote changes on the current branch into the local clone, and merges them into the current working files.

Fetch command: Downloads changes from a remote repository into the local clone.

Checkout command: Replaces the current working files with files from a branch.


```

MINGW64 ~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git fetch https://github.com/beawareofme/Cognizance.git
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 8 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 1.30 KiB | 33.00 KiB/s, done.
From https://github.com/beawareofme/Cognizance
 * branch            HEAD       -> FETCH_HEAD

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git pull
From github.com:beawareofme/Cognizance
Merge made by the 'ort' strategy.
 welcome.txt | 1 +
 hi.txt      | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 hi.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push -u origin master
Enumerating objects: 15, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 889 bytes | 444.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To github.com:beawareofme/Cognizance.git
 1d55920..e2e2a67 master -> master
branch 'master' set up to track 'origin/master'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ 1
bash: 1: command not found

```

```

MINGW64 ~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
Task1 welcome.txt bye.c hello.py hi.txt index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout contact
error: pathspec 'contact' did not match any file(s) known to git

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout Contact
error: pathspec 'Contact' did not match any file(s) known to git

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git fetch origin
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 689 bytes | 114.00 KiB/s, done.
From github.com:beawareofme/Cognizance
 * [new branch]      contact -> origin/contact

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git branch
  Test1
  Test2
* master

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout Contact
Switched to a new branch 'Contact'.
branch 'Contact' set up to track 'origin/Contact'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ git branch
 * Contact
  Test1
  Test2
  master

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ git checkout master
Switched to branch 'master'

```

```

branch 'Contact' set up to track 'origin/Contact'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ git branch
 * Contact
  Test1
  Test2
  master

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git branch
  Contact
  Test1
  Test2
* master

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$

```

Merge: Merges files from a given branch into the current branch.

```

MINGW64 ~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout Contact
Switched to branch 'Contact'
Your branch is up to date with 'origin/Contact'.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ touch m1.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ touch m2.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (Contact)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git rm f1.txt
fatal: pathspec 'f1.txt' did not match any files

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
Task1 Welcome.txt bye.c f1.txt f2.txt hello.py hi.txt index.html m1.txt m2.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git rm f1.txt
fatal: pathspec 'f1.txt' did not match any files

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git merge Contact
Merge made by the 'ort' strategy.
  contact.txt | 1 +
  1 file changed, 1 insertion(+)
  create mode 100644 contact.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
Task1 Welcome.txt bye.c contact.txt f1.txt f2.txt hello.py hi.txt index.html m1.txt m2.txt

```

```

MINGW64 ~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 3 commits.
  (use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  f1.txt
  f2.txt
  m1.txt
  m2.txt

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

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add -A

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git commit -a -m "idk what I am doing"
[master ebe512c] idk what I am doing
 4 files changed, 2 insertions(+)
  create mode 100644 f1.txt
  create mode 100644 f2.txt
  create mode 100644 m1.txt
  create mode 100644 m2.txt

```

Rebase: Integrates changes from one branch into another. It is an alternative to the better known “merge” command.

```

MINGW64 ~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout test
Switched to branch 'test'

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ touch a1.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ code .

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git push
git@github.com: Permission denied (publickey).
fatal: could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  a1.txt

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

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout test
Switched to branch 'test'

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git status

```

```

MINGW64~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git status
On branch test
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    a1.txt

nothing added to commit but untracked files present (use "git add" to track)
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git add -A
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git commit -a -m "rebase practice"
[test 3d6290e] rebase practice
1 file changed, 1 insertion(+)
create mode 100644 a1.txt
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git b1.txt
git: 'b1.txt' is not a git command. See 'git --help'.
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ code .
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch b1.txt
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    b1.txt

b1.txt
nothing added to commit but untracked files present (use "git add" to track)
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add -A
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git commit -a -m "hello"
[master a575ee6] hello
1 file changed, 1 insertion(+)
create mode 100644 b1.txt
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout tets
error: pathspec 'tets' did not match any file(s) known to git
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git checkout test
Switched to branch 'test'
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ ls
Task1 Welcome.txt a1.txt bye.c contact.txt f1.txt f2.txt hello.py hi.txt index.html m1.txt m2.txt
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ git rebase master
Successfully rebased and updated refs/heads/test.
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$ ls
Task1 Welcome.txt a1.txt b1.txt bye.c contact.txt f1.txt f2.txt hello.py hi.txt index.html m1.txt m2.txt
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (test)
$

```

Reset: Makes the current branch point to some specific revision or branch

```

MINGW64 ~/Users/prans/Cognizance
prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch example1

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ touch example2

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add -A

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   aud1.txt
        new file:   example1
        new file:   example2
        modified:   index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git reset
Unstaged changes after reset:
M       index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git statud
git: 'statud' is not a git command. See 'git --help'.

The most similar command is
    status

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

Changes not staged for commit:

```

```

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        aud1.txt
        example1
        example2

no changes added to commit (use "git add" and/or "git commit -a")

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git add -A

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   aud1.txt
        new file:   example1
        new file:   example2
        modified:   index.html

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$

```

Clone: Clone an existing repository into a new directory.

```

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
task1 welcome.txt aud1.txt bl.txt bye.c contact.txt example1 example2 fl.txt f2.txt hello.py hi.txt index.html n1.txt n2.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ git clone https://github.com/illuminate/html.git
Cloning into 'html'...
remote: Enumerating objects: 609, done.
remote: Total 609 (delta 0), reused 0 (delta 0), pack-reused 609
Receiving objects: 100% (609/609), 98.19 KiB | 683.00 KiB/s, done.
Resolving deltas: 100% (260/260), done.

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$ ls
task1 welcome.txt aud1.txt bl.txt bye.c contact.txt example1 example2 fl.txt f2.txt hello.py hi.txt html/ index.html n1.txt n2.txt

prans@DESKTOP-KD3PIPF MINGW64 ~/Cognizance (master)
$

```

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

