**Assignment 1**

sdabekar@LIN25006369 MINGW64 ~

$ mkdir project\_dir

sdabekar@LIN25006369 MINGW64 ~

$ cd project\_dir

sdabekar@LIN25006369 MINGW64 ~/project\_dir

$ git init

Initialized empty Git repository in C:/Users/SDABEKAR/project\_dir/.git/

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git status

On branch master

No commits yet

Untracked files:

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

index.html.txt

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

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add index.html

fatal: pathspec 'index.html' did not match any files

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add index.html

fatal: pathspec 'index.html' did not match any files

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add .

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: index.html.txt

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git commit -m "index.html file created"

[master (root-commit) a6d3da7] index.html file created

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 index.html.txt

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git status

On branch 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: index.html.txt

Untracked files:

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

info.txt.txt

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

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git config

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 <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

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ touch .gitignore

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git status

On branch 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: index.html.txt

Untracked files:

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

.gitignore

info.txt.txt

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

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add index.html

fatal: pathspec 'index.html' did not match any files

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git status

On branch 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: index.html.txt

Untracked files:

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

.gitignore

info.txt.txt

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

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add .

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: .gitignore

modified: index.html.txt

new file: info.txt.txt

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git log

commit a6d3da70d6fdd420d1e679a9a0b5244c439053c2 (HEAD -> master)

Author: SDABEKAR <SDABEKAR@LIN25006369>

Date: Wed Dec 28 16:53:00 2022 +0530

index.html file created

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git log --oneline

a6d3da7 (HEAD -> master) index.html file created

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add index.html

fatal: pathspec 'index.html' did not match any files

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add .A

fatal: pathspec '.A' did not match any files

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git add .

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git revert

usage: git revert [--[no-]edit] [-n] [-m <parent-number>] [-s] [-S[<keyid>]] <commit>...

or: git revert (--continue | --skip | --abort | --quit)

--quit end revert or cherry-pick sequence

--continue resume revert or cherry-pick sequence

--abort cancel revert or cherry-pick sequence

--skip skip current commit and continue

--cleanup <mode> how to strip spaces and #comments from message

-n, --no-commit don't automatically commit

-e, --edit edit the commit message

-s, --signoff add a Signed-off-by trailer

-m, --mainline <parent-number>

select mainline parent

--rerere-autoupdate update the index with reused conflict resolution if possible

--strategy <strategy>

merge strategy

-X, --strategy-option <option>

option for merge strategy

-S, --gpg-sign[=<key-id>]

GPG sign commit

--reference use the 'reference' format to refer to commits

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git config --global alias.my-add add

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git my-add .

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git commit -m "index.html updated and rename the add command"

[master ab4caf3] index.html updated and rename the add command

3 files changed, 13 insertions(+)

create mode 100644 .gitignore

create mode 100644 info.txt.txt

sdabekar@LIN25006369 MINGW64 ~/project\_dir (master)

$ git revert

usage: git revert [--[no-]edit] [-n] [-m <parent-number>] [-s] [-S[<keyid>]] <commit>...

or: git revert (--continue | --skip | --abort | --quit)

--quit end revert or cherry-pick sequence

--continue resume revert or cherry-pick sequence

--abort cancel revert or cherry-pick sequence

--skip skip current commit and continue

--cleanup <mode> how to strip spaces and #comments from message

-n, --no-commit don't automatically commit

-e, --edit edit the commit message

-s, --signoff add a Signed-off-by trailer

-m, --mainline <parent-number>

select mainline parent

--rerere-autoupdate update the index with reused conflict resolution if possible

--strategy <strategy>

merge strategy

-X, --strategy-option <option>

option for merge strategy

-S, --gpg-sign[=<key-id>]

GPG sign commit

--reference use the 'reference' format to refer to commits

**Assignment 2**

sdabekar@LIN25006369 MINGW64 ~

$ mkdir Assignments

sdabekar@LIN25006369 MINGW64 ~

$ cd Assignments

sdabekar@LIN25006369 MINGW64 ~/Assignments

$ nano Readme.txt

sdabekar@LIN25006369 MINGW64 ~/Assignments

$ git add Readme.txt

fatal: not a git repository (or any of the parent directories): .git

sdabekar@LIN25006369 MINGW64 ~/Assignments

$ git init

Initialized empty Git repository in C:/Users/SDABEKAR/Assignments/.git/

sdabekar@LIN25006369 MINGW64 ~/Assignments (master)

$ git add Readme.txt

warning: in the working copy of 'Readme.txt', LF will be replaced by CRLF the next time Git touches it

sdabekar@LIN25006369 MINGW64 ~/Assignments (master)

$ git commit -m "first"

[master (root-commit) b1c66eb] first

1 file changed, 1 insertion(+)

create mode 100644 Readme.txt

sdabekar@LIN25006369 MINGW64 ~/Assignments (master)

$ git checkout -b html-assignments

Switched to a new branch 'html-assignments'

sdabekar@LIN25006369 MINGW64 ~/Assignments (html-assignments)

$ git commit -m "second"

On branch html-assignments

nothing to commit, working tree clean

sdabekar@LIN25006369 MINGW64 ~/Assignments (html-assignments)

$ git push origin html-assignments

fatal: 'origin' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

sdabekar@LIN25006369 MINGW64 ~/Assignments (html-assignments)

$ git push -u origin master

fatal: 'origin' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

sdabekar@LIN25006369 MINGW64 ~/Assignments (html-assignments)

$ git remote add origin https://github.com/SujitaD/GithubAssignment.git

sdabekar@LIN25006369 MINGW64 ~/Assignments (html-assignments)

$ git push origin html-assignments

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

Writing objects: 100% (3/3), 218 bytes | 54.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

remote:

remote: Create a pull request for 'html-assignments' on GitHub by visiting:

remote: https://github.com/SujitaD/GithubAssignment/pull/new/html-assignments

remote:

To https://github.com/SujitaD/GithubAssignment.git

\* [new branch] html-assignments -> html-assignments