Configuration

S R PANDA@SATYA MINGW64 ~

$ git config --global user.name "srpanda"

S R PANDA@SATYA MINGW64 ~

$ git config --global user.email "srpcode10@gmail.com"

S R PANDA@SATYA MINGW64 ~

$ git --version

git version 2.41.0.windows.3

S R PANDA@SATYA MINGW64 ~

$ 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/etc/ssl/certs/ca-bundle.crt

core.autocrlf=true

core.fscache=true

core.symlinks=false

core.fsmonitor=true

pull.rebase=false

credential.helper=manager

credential.https://dev.azure.com.usehttppath=true

init.defaultbranch=master

user.name=srpanda

user.email=srpcode10@gmail.com

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

If you want to use your code editor 🡪 default is VS Code

S R PANDA@SATYA MINGW64 ~

$ git config --global core.editor "code"

Open your VS Code IDE or access the VS Code

S R PANDA@SATYA MINGW64 ~

$ code

Change Directory

$ cd Desktop

Check List of contents inside directory

S R PANDA@SATYA MINGW64 ~/Desktop

$ ls

Create a “Project” Folder Inside Desktop Directory

S R PANDA@SATYA MINGW64 ~/Desktop

$ mkdir projects

Change Directory to “Project” Folder/If you want to access this folder 🡪

S R PANDA@SATYA MINGW64 ~/Desktop

$ cd projects

Inside “Project” Folder you have to create a “digital\_clock” Folder

S R PANDA@SATYA MINGW64 ~/Desktop/projects

$ mkdir digital\_clock

Change Directory to “digital\_clock” Folder

S R PANDA@SATYA MINGW64 ~/Desktop/projects

$ cd digital\_clock

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock

$ ls

Check Hidden Files Inside Directory

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock

$ ls –a

./ ../

initialize a new Git repository in a directory

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock

$ git init

Initialized empty Git repository in C:/Users/S R PANDA/Desktop/projects/digital\_clock/.git/

Note: The command "git init" is used to initialize a new Git repository in a directory. When you run this command, Git creates a new repository in the current directory, and it starts tracking changes for all the files in that directory.

Git Status Check:

$ git status

What is Branch?

Branch can be said like It’s an independent snapshots of repository and each repository can have multiple branches.

How to create a files inside repository 🡪 using touch command.

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ touch file-1.txt

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ touch file-2.txt

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ ls

file-1.txt file-2.txt

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git status

On branch master

No commits yet

Untracked files:

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

file-1.txt

file-2.txt

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

List of files in the staging area

The command "git ls-files" is used to list the files currently being tracked by Git in the repository. It displays the names of the files that are staged or committed in the working tree.

When you run this command, Git will list the names of all the files in the current repository that are being tracked. If there are no files being tracked, the command will not display anything.

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git ls-files

Add changes to the staging area in Git.

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git add file-1.txt file-2.txt

**Check Status Again –**

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git status

On branch master

No commits yet

Changes to be committed:

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

new file: file-1.txt

new file: file-2.txt

**If you want to remove the specific files from staging area or you don’t want to commit it**

**then you just write 🡪**

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git rm --cached file-2.txt

rm 'file-2.txt'

**If you want to add again these files or files that present in your local directory in the staging area 🡪**

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git add file-2.txt

**If you want to add all files inside on your local repository 🡪**

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git add .

Commit Change to git Repo

The commit message will be "added two new dummy files."

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git commit -m "added two new dummy files"

[master (root-commit) 02044fc] added two new dummy files

2 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 file-1.txt

create mode 100644 file-2.txt

Create a new files

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ touch digitalClock.py

Open Default Editor i.e. VS Code

In VS Code I want to open my files i.e. digitalClock.py

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ code digitalClock.py

In that Python file, I have just created a small python program using VS Code IDE.

Then I Run it through Git or

Execute the python file

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ python digitalClock.py

Add my files to staging area.

Then I Passed the commit message ..

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git add .

Put a commit message 🡪

S R PANDA@SATYA MINGW64 ~/Desktop/projects/digital\_clock (master)

$ git commit -m "added a new digital clock program"

[master b583b80] added a new digital clock program

1 file changed, 7 insertions(+)

create mode 100644 digitalClock.py

Git Workflow

Working Directory

Staging Area

Git Repository

ADD COMMIT

Local Directory/Repository

Git Log

It shows basically history of commit that you have input.

When executed in a Git repository, it displays a chronological list of commits made in that repository. Each commit entry typically shows information such as the commit hash, author name, date, and commit message.

$ git log

commit b583b80208da6278e760efaad7388b9cbaee8c5f (HEAD -> master)

Author: srpanda <srpcode10@gmail.com>

Date: Fri Aug 4 03:50:03 2023 +0530

added a new digital clock program

HEAD -> master

It is pointing to the current branch.

commit 02044fcf512160d785b62eb0f1870323fda6ea85

Author: srpanda <srpcode10@gmail.com>

Date: Fri Aug 4 03:31:30 2023 +0530

added two new dummy files

if you want to display your commit in single line

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git log –oneline

b583b80 (HEAD -> master) added a new digital clock program

02044fc added two new dummy files

Git Branch

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git branch

\* master

Create a new branch

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git branch clock

See the list of branch that exit in your git.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git branch

clock

\* master

Here star is pointing to the current branch i.e. master.

How can you switch back from one branch to another 🡪

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git switch clock

Switched to branch 'clock'

"master” branch and “clock” branch , these are 2 different version.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (clock)

$ git log –oneline

892dee1 (HEAD -> clock) added digital clock

dded2a9 (master) remove welcome message

4d3137b added a welcome message

b583b80 added a new digital clock program

02044fc added two new dummy files

Whatever I have made changes in “master” branch that will not reflect to “clock” branch and vice-versa.

Because these are two different version of my project.

They are containing different types of code of same project.

“master” branch version of my project

Switch from “clock” branch to “master” branch.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (clock)

$ git switch master

Switched to branch 'master'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ python digitalClock.py

“master” branch belonging this code

from tkinter import Tk, Label

window = Tk()

window.title("Digital Clock")

window.geometry("600x300")

window.configure(bg = "steelblue")

label = Label(window, font = ("Arial Black",78,"bold"), bg = "steelblue" ,fg = "white")

label.pack(pady = 100)

window.mainloop()



“clock” branch version

Switch from “master” branch to “clock” branch.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git switch clock

Switched to branch 'clock'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (clock)

$ python digitalClock.py

“clock” branch belonging this code

from tkinter import Tk, Label

from datetime import datetime

window = Tk()

window.title("Digital Clock")

window.geometry("600x300")

window.configure(bg = "steelblue")

label = Label(window, font = ("Arial Black",78,"bold"), bg = "steelblue" ,

fg = "white")

label.pack(pady = 100)

def clock():

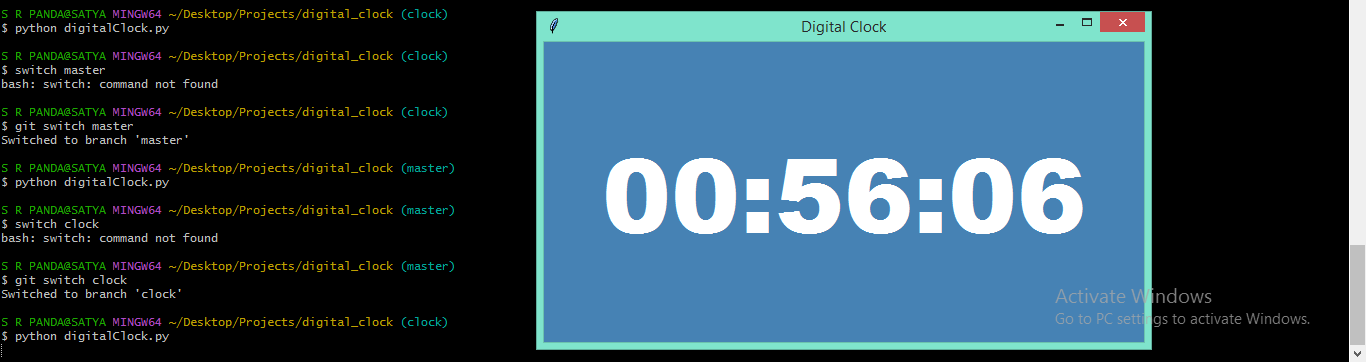
    time = datetime.now().strftime("%H:%M:%S")

    label.configure(text = time)

    label.after(500,clock)

clock()

window.mainloop()



Merging Branch

I want to merge “clock” branch of code with “master” branch.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (clock)

$ git switch master

Switched to branch 'master'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git log --oneline

dded2a9 (HEAD -> master) remove welcome message

4d3137b added a welcome message

b583b80 added a new digital clock program

02044fc added two new dummy files

It will merge all the code or all the files from “clock” branch to “master” branch.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git merge clock

Updating dded2a9..892dee1

Fast-forward

digitalClock.py | 7 +++++++

1 file changed, 7 insertions(+)

Now my “master” branch having “digital clock”.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git log --oneline

892dee1 (HEAD -> master, clock) added digital clock

dded2a9 remove welcome message

4d3137b added a welcome message

b583b80 added a new digital clock program

02044fc added two new dummy files

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git switch clock

Switched to branch 'clock'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (clock)

$ git log --oneline

892dee1 (HEAD -> clock, master) added digital clock

dded2a9 remove welcome message

4d3137b added a welcome message

b583b80 added a new digital clock program

02044fc added two new dummy files

Now everything looks same.

Git Reset

If you want go back the previous commit –

Or if you want to go back your old history 🡪

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git log –oneline

892dee1 (HEAD -> master, clock) added digital clock

dded2a9 remove welcome message

4d3137b added a welcome message

b583b80 added a new digital clock program

02044fc added two new dummy files

These are called hash code of commit

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git reset --hard b583b80

HEAD is now at b583b80 added a new digital clock program

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git log --oneline

b583b80 (HEAD -> master) added a new digital clock program

02044fc added two new dummy files

Git Detached Head State

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git checkout 4d3137b

Note: switching to '4d3137b'.

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 4d3137b added a welcome message

This branch is now a temporary branch.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((4d3137b...))

$ git log –oneline

4d3137b (HEAD) added a welcome message

b583b80 (master) added a new digital clock program

02044fc added two new dummy files

I have changed something in my code.

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((4d3137b...))

$ code digitalClock.py

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((4d3137b...))

$ python digitalClock.py

from tkinter import Tk, Label

window = Tk()

window.title("Digital Clock")

window.geometry("600x300")

window.configure(bg = "red")

label = Label(window,text = "Welcome!", font = ("Arial Black",78,"bold"), bg = "red" ,

fg = "white")

label.pack(pady = 100)

window.mainloop()



Commit this code

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((4d3137b...))

$ git add .

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((4d3137b...))

$ git commit -m "change to red screen"

[detached HEAD 830857b] change to red screen

1 file changed, 2 insertions(+), 2 deletions(-)

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((830857b...))

$ git log –oneline

830857b (HEAD) change to red screen

4d3137b added a welcome message

b583b80 (master) added a new digital clock program

02044fc added two new dummy files

Save this changes to new branch

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock ((830857b...))

$ git switch -c RED\_Screen

Switched to a new branch 'RED\_Screen'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (RED\_Screen)

$ git branch

\* RED\_Screen

clock

master

How my 3 branches are working ..

RED\_Screen Branch

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (RED\_Screen)

$ git status

On branch RED\_Screen

nothing to commit, working tree clean

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (RED\_Screen)

$ python digitalClock.py



Master Branch

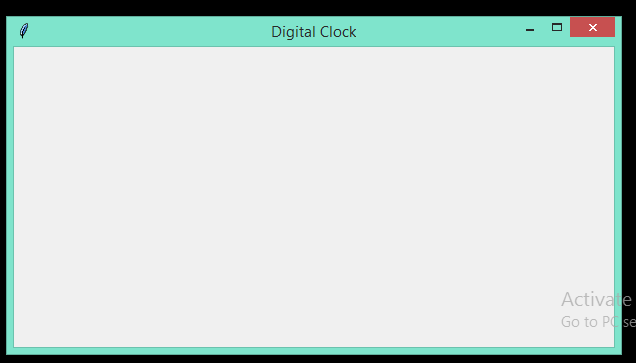
S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (RED\_Screen)

$ git switch master

Switched to branch 'master'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ python digitalClock.py



Clock branch

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (master)

$ git switch clock

Switched to branch 'clock'

S R PANDA@SATYA MINGW64 ~/Desktop/Projects/digital\_clock (clock)

$ python digitalClock.py



So, basically in same repository I am working 3 branches that consist of 3 version of my projects or code.

That means, each branches having own version of code.

If you want, then you can merge with single branch of code.