**What is git?** Source Control Manegment(SCM)

**How to congigure name of the user:**git config –global user.name “Charbel Hajj Moussa”

**How to specify email address:**git config –global user.email [charbelhajjmoussa7@gmail.com](mailto:charbelhajjmoussa7@gmail.com)

**How to configure default branch name:**git config –global init.default branch main

**For help**

gitconfig -h

or git help config

**change directory**

cd and path

**To initialize a new reporitory** (by default all the file inside the file will be untracked)

git init

**Status od our repository**

git status

**To track a file we can**

git add example.html (this will be tracked)

clear(to clear console)

**And to re remove this file**

Git rm –cached <example.html>

**To create some files that are private and I wónt to any won access fro git I have to create a file .gitignore and then edit it with notepad and insert:**#ignore all .txt files

\*.txt

**To track all the files :**

git add –all (or git add . or git add -A)

**To commit is to create a save point:**git commit -m “this is a message”

**If I want to see the different before and after update of a file :**

git diff (so now I can compare the 2 version)

**You have 3 main parts:**

1. **Working files**
2. **Statging(LIKE A PIN)**
3. **Commit**

**So when you work you ave to push your file in the stage**

git add index.htm (SO NOW IF I COMMIT THIS FILE WILL BE SAVED)

**And now let’s return it to the working file :**

git restore –staged index.html**(so noew if we commit this file will not be saved)**

**And now if I want to save all the files in the working and statfing we can use:**

git commit -a -m “updated text to free range”

**we can delete a file from our repo with the command:**

git rm “example.htm”

**And we can restor this file :**

git restore “example.htm”

**we can rename any file in our repo:**

git mv “OLD.png” “Newname.png”

**We can see al the commits that we made:**

git log

**We can see a summary of the commits already made:**

git log --oneline

**We can rename the last commit :**

git commit -m “Changed the name of the last commit”

**We can see the more specific details of the commits already done:**

git log -p(Q for exit from this mode)

**To goback to any commit**

git reset …(mahal … menhot code l commit)

**To change the view of logs**

git rebase -i --root

**Let’s create a new branch**

git branch ExampleBranch

**To see all the branches we can use :**

git branch

**To seitch to another branch we can use:**

git switch ExampleBranch

**And now let’s switch back to the main**

git switch main

**And now we can merge the ExampleBranch to the main :**  
get merge -m “Merge ExampleBranch back to main” ExampleBranch **(So now we have evrithing that we edited in the ExampleBranch in the Main branch)**

**To delete** ExampleBranch :

git branch -d ExampleBranch

**we can create and directly siwcth to a new branch:**

git switch -c UpdateText

**So now if we change the first word in a file A in the UpdateText branch and then we switch back to the main branch and also we change the same word in the same file andcommit when we want to merge back the files from the UpdateText to the main then we give an error of conflict and put you in the (main|merging) then you have to open this file and choose one version from 2 and save it then commit.**

**Online:**

**To create a new repo:**

**echo "# TESTLYNO" >> README.md**

**git init**

**git add README.md**

**git commit -m "first commit"**

**git branch -M main**

**git remote add origin https://github.com/CHM2023/TESTLYNO.git**

**git push -u origin main**

**to push an existing repo**

**git remote add origin https://github.com/CHM2023/TESTLYNO.git**

**git branch -M main**

**git push -u origin main**