

## Pràctica 2) Fonaments bàsic de funcionament del sistema de control de versions Git. Ordres per gestionar el SCV Git.

### DOCUMENTACIÓ

- 1- Conceptes bàsics de Git: Snapshots. Integritat. Operacions locals. Les 3 àrees de residència dels fitxers.  
<http://git-scm.com/book/en/Getting-Started-Git-Basics>  
<http://git-scm.com/book/es/Empezando-Fundamentos-de-Git>
- 2- El cicle de vida dels fitxers: Enregistrant canvis en el dipòsit.  
<http://git-scm.com/book/en/Git-Basics-Recording-Changes-to-the-Repository>  
<http://git-scm.com/book/es/Fundamentos-de-Git-Guardando-cambios-en-el-repositorio>
- 3- Fonament bàsics de Git  
[Llibre Pro Git en anglès. Fonaments bàsics de Git --> capítol 2](#)  
[Llibre Pro Git en castellà. Fonaments bàsics de Git --> capítol 2](#)
- 4- Treballant amb ordres
  - a) Tutorial pràctic en línia de Github i ordres Git: <https://try.github.io/levels/1/challenges/1>
  - b) Explicació resumida d'algunes de les ordres més importants: <http://rogerdudler.github.io/git-guide/>
  - c) Configuració inicial: <http://git-scm.com/book/en/Getting-Started-First-Time-Git-Setup>

### 1- Conceptes bàsics

- a) Què és un snapshot o instantània?
- b) Quan GIT realitza una instantània, que fa amb els fitxers que no han estat modificats?
- c) Què és i quina és la utilitat de la suma de comprovació o checksum?. Quin és el seu format?.
- d) Quins són els tres estats en els quals es pot trobar un arxiu d'un projecte?.
- e) Què hi ha al "Working directory" (o directori de treball)?
- f) Què hi ha a la "Staging area" (o area de preparació). A on es troba?
- g) Què hi ha al "Git directory" (o directori Git). A on es troba?
- h) Quina acció realitzem per passar de un fitxer del directori de treball a l'àrea de preparació?
- i) Quina acció realitzem per passar de un fitxer de l'àrea de preparació al directori Git?
- j) Quan realitzem una instantània (o snapshot)?
- k) Què és un fitxer "untracked" (no seguit o no controlat)?
- l) Què és un fitxer "tracked" (seguit o controlat)?
- m) En quins estats es pot trobar un fitxer "tracked"?
- n) Quina diferència hi ha entre un fitxer modificat i un fitxer en preparació?
- o) Què hem de fer per poder "consignar" (commit) d'un fitxer "sense seguiment" (untracked)?
- p) Un fitxer modificat, si no està en preparació (staged) pot quedar consignat (commit)?
- q) Després de consignar tots els arxius d'un projecte, quin és el seu estat?
- r) Quin és l'estat de tots els arxius d'un projecte quan es fa un clonatge des d'un dipòsit per primera vegada?

### 2- Treballant amb ordres per gestionar el SCV Git

- a) Crea un fitxer de configuració mínima de GIT pel teu usuari amb la informació que trobaràs a **4.c**. Fes que a la configuració surti com a nom d'usuari, el nom d'usuari **Guthub** que vas crear a la pràctica **m08uf1pr1b**, i com a correu electrònic, el compte de **Gmail** que vas crear a la pràctica **m08uf4pr1a**. Executa l'ordre per comprovar la nova configuració. Utilizat com editor predefinit **nano**. Comprova que s'ha creat el fitxer **.gitconfig** al teu **directori personal**. Si hi ha cap error, pots esborrar una entrada amb la informació que trobaràs [aquí](#) (Mira la primera resposta que treballa amb l'opció **--unset**)
- b) Crea dins del teu directori personal un nou directori de nom **m08uf4pr2**. Dins d'aquest directori, crea un altre directori de nom **mitjana**. Des d'aquest últim directori, descarrega el fitxers **mitjana.c** i **README** executant:

```
wget http://www.collados.org/daw2/m08/uf4/mitjana.c
```

```
wget http://www.collados.org/daw2/m08/uf4/README
```

- c) Crea i inicialitza un nou dipòsit local Git pels fitxers del projecte **mitjana**. Comprova que s'ha creat el directori **.git** dins del directori **mitja**. Comprova l'estat del dipòsit. Indica quin és l'estat dels fitxers **mitjana.c** i

**README.** Explica el motiu pel qual es troben en aquest estat.

**d)** Afegeix (**add**) els dos fitxers a l'àrea de preparació (staged area). Comprova ara l'estat del dipòsit. Indica quin és l'estat dels fitxers **mitjana.c** i **README**.

**e)** Modifica el fitxer **mitjana.c**. A la línia 16, canvia **scanf("%c",&consulta);** per **scanf("%s",&consulta);**. Comprova ara l'estat del dipòsit. Indica quin és l'estat del fitxer **mitjana.c**. D'acord amb l'estat actual, en el moment de fer una consignació (**commit**), quina versió de l'arxiu s'emmagatzemarà en el dipòsit GIT?.

**f)** Consigna (**commit**) els fitxers que es troben a l'àrea de preparació (staged area). En el moment de consignar afegeix el comentari "*Primer commit del projecte mitjana*". Troba el valor del **checksum** (o suma de comprovació) que s'ha creat per aquesta consignació (**commit**). Comprova la llista de fitxers afegits.

**g)** Comprova ara l'estat del dipòsit. Indica quin és l'estat dels fitxers **mitjana.c** i **README**. Quin és l'estat de cada fitxer?.

**h)** Afegeix **mitjana.c** a l'àrea de preparació i fes una nova consignació. En el moment de consignar afegeix el comentari "*Segon commit del projecte mitjana*". Troba el nou valor de la nova suma de comprovació. Comprova novament l'estat del dipòsit. Indica el motiu del resultat mostrat per pantalla.

**i)** Compila el programa **mitjana.c** amb l'ajut del **README**. Comprova l'estat del dipòsit Git. Quin és l'estat del programa **mitjana**? Per què?.

**j)** Explica quin és el propòsit del fitxer **.gitignore**. Crea un fitxer de nom **.gitignore** dins del directori **mitjana** amb el següent contingut:

```
*.[sio]
*~
.gitignore
mitjana
```

A continuació compila el programa **mitja.c** amb els següents paràmetres: **gcc mitja.c -save-temps -o mitjana**. Comprova els arxius que hi han ara al teu directori. Comprova l'estat del dipòsit. Explica perquè no hi ha res per afegir i consignar.

**k)** Afegeix el fitxer **mitjana** a l'àrea de preparació. Ho pots fer?. Per què?. Comprova l'estat del dipòsit.

**l)** Modifica el fitxer **mitjana.c**. A la línia 4, canvia **flotat suma=0;** per **flotat suma=0.0;**. Comprova ara l'estat del dipòsit i del fitxer **mitjana.c**. Afegeix el fitxer **mitjana.c** a l'àrea de preparació. Comprova ara l'estat del dipòsit i del fitxer **mitjana.c**.

**m)** Torna a modificar el fitxer **mitjana.c**. A la línia 5, canvia **float mitjana;** per **float mitjana=0.0;**. Comprova l'estat del dipòsit.

**n)** Presenta per pantalla un historial de totes les consignacions realitzades. Comprova el valor del **checksum** de tots els commits.

**o)** Fes un **add** del fitxer **mitjana.c** i després un **commit**. En el moment de consignar afegeix el comentari "*Tercer commit del programa mitjana*". Comprova l'estat del projecte. Comprova també l'historial. Comprova a quina branca estas del projecte.

**p)** Crea un projecte **Públic** de nom **mitjana** al teu dipòsit personal de Github. Veuràs que en el moment de la creació del nou dipòsit et donen informació de com utilitzar-lo. Llegeix la segona recomanació (..or push an existing repository from the command line) i fes un **push** del teu projecte local al dipòsit remot. Primer hauràs d'afegir el dipòsit remot a la llista de dipòsits remots amb els quals treballes fent un **remote add** i després faràs el **push**. Un cop hagis acabat, comprova que has pujat el projecte al teu dipòsit remot. Què és **origin**? --> Ajut: <http://stackoverflow.com/questions/9529497/what-is-origin-in-git>.

**q)** Crea una nova carpeta de nom **clone\_m08uf4pr2** dins del teu directori personal. **Clona** (Ajut: <https://git-scm.com/docs/git-clone>) dins d'aquesta carpeta el projecte **mitjana**. Comprova que s'ha creat la carpeta **mitjana** i que dins d'aquesta carpeta hi ha un nou dipòsit local. Comprova l'estat del dipòsit clonat. Comprova l'historial.

**r)** Modifica el fitxer **mitjana.c** que es troba a **~/clone\_m08uf4pr2/mitjana**. A la línia 6, canvia **float nou\_valor;** per **float nou\_valor=0.0;**. Fes un **add** i un **commit**. En el moment de consignar afegeix el comentari "*Quart commit del programa mitjana*". Puja el canvi (fes un **push**) al dipòsit remot. Tingués en compte que com has clonat el dipòsit no cal afegir el dipòsit remot a llista de manera que només cal fer el **push**. Comprova des de Github l'estat del dipòsit remot.

**s)** Torna a **~/m08uf4pr2/mitjana** (el directori original). Baixa (fes un **pull**) del projecte que està al dipòsit remot al dipòsit local. Ara obre el fitxer **mitjana.c** i comprova que s'ha afegit el nou codi. Ajut: <https://try.github.io/levels/1/challenges/12>.

**t)** Modifica el fitxer **mitjana.c** que es troba a **~/m08uf4pr2/mitjana**. Canvia a la línia 8 **char consulta='s';** per **consulta='y';**. Afegeix el fitxer **mitjana.c** a l'àrea de preparació. Modifica a la línia 10 **while(consulta=='s')** per **while(consulta=='y')**. Comprova les següents diferències: **a)** La diferència entre la informació al directori de treball i l'últim **commit** i **b)** La diferència entre la informació a l'àrea de prep directori de treball i

l'últim **commit**.

u) Esborra el fitxer **mitjana.c** de l'últim **add** realitzat. Comprova l'estat del projecte.

v) Fes que el fitxer **mitjana.c** que es troba al directori de treball torni a utilitzar el codi que hi havia a l'últim **commit** realitzat. Comprova l'estat del projecte. Comprova el contingut de **mitjana.c**.

w) Reverteix el fitxer **mitjana.c** que es troba a l'àrea de treball a la versió que tenia al **2n commit** començat des del darrere. Comprova el contingut de **mitjana.c**.

x) Comprova l'historial de commits realitzats fins aquest punt. Ara esborra completament l'últim **commit** realitzat a la branca **master** amb l'ordre:

```
git reset --hard HEAD~1
```

Visualitza ara l'historial de commits realitzats i comprova que l'últim **commit** ha estat esborrat.

y) Indica quin és el significat del següent avis:



### Lliurament de la pràctica

#### Forma de lliurament de l'informe de la pràctica (75%)

1- Treball individual.

2- El nom del fitxer ha de tenir el següent format **daw2\_cognom1\_nom\_m08uf4pr2.pdf**

3- Envieu la solució proposada per correu electrònic:

Adreça: **cf@collados.org**

Assumpte: **daw2\_cognom1\_nom\_m08uf4pr2**

4- La **data límit** de lliurament de la pràctica és el dia **4-11-18** a les **23.59 hores**. Fora d'aquest límit **NO** acceptaré cap pràctica.

5- Format Lletra: **Arial 10**. Marges (superior, inferior, esquerra, dreta): **2cm**

6- Numeració de pàgina: **Peu de pàgina a la dreta**.

#### Forma de lliurament de la part pràctica (25%)

1- La comprovacions es farà els dia **5-11-17**

2- Lliurament:

a) Crea una carpeta de nom **m08uf4pr2pp** dins del teu directori personal.

**b)** Des de dins de la carpeta **m08uf4pr2pp** clona el projecte bomba que trobaràs a **<https://github.com/dc1415daw2/bomba.git>**.

**c)** Modifica el fitxer **main.c**. Afegeix darrere de la **línia 11** les següents instruccions:

```
printf("PATABOOOUUUUMMMM\n");  
printf("CAXIPUUUUUMMMM\n");
```

**Afegeix** els fitxers del projecte a l'àrea de preparació, i després fes un **commit**. En el moment de fer el commit, afegeix el comentari "Versió 2 del projecte bomba".

**d)** Per fer la comprovació final, executa:

```
git status  
git log  
git remote -v  
cat bomba.c
```

**e)** Compila i executa **main.c** d'acord amb el **Makefile** inclòs dins del projecte.