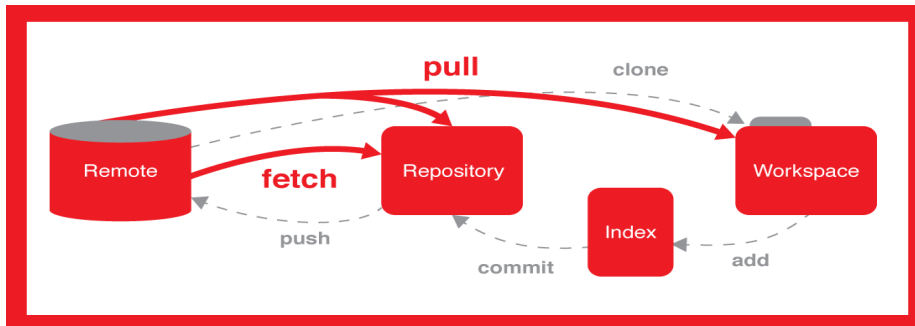


Pràctica 5) Sincronitzant el dipòsit remot amb el local

Conceptes previs

a) pull i fetch

- Esquema:



NOTA 1: Dins del Repository Git té a localitzacions diferents les branques locals i remotes.

NOTA 2: Per fer que les branques locals tinguin el contingut de les branques remotes s'han de fusionar

NOTA 3: Quan clonem o fem un pull es baixen les branques del servidor remot i es fusionen amb les branques locals. També fem que el directori de treball estigui sincronitzat amb el dipòsit local.

NOTA 4: Quan fem un fetch no fusionem i no sincronitzem el directori de treball

NOTA 5: fetch és segur perquè no canvia el dipòsit local i tampoc el directori de treball.

- Diferències: <https://www.git-tower.com/learn/git/faq/difference-between-git-fetch-git-pull>
- git fetch: <https://aulab.es/articulos-guias-avanzadas/90/el-comando-git-fetch-en-git>

b) HEAD vs main vs origin

- Breu explicació: <https://stackoverflow.com/questions/8196544/what-are-the-git-concepts-of-head-master-origin> (Veure la primera resposta d'en Matt Greer).
- Una altra explicació: <https://www.campusmvp.es/recursos/post/git-los-conceptos-de-master-origin-y-head.aspx>

Part 1: Fent commits i sincronitzant dipòsits utilitzant git pull

1- Crea una carpeta de projecte de nom **html2pdf**. Accedeix a la carpeta i **inicialitza** un dipòsit.

2- Afegeix (i verifica) els remotes per accedir al dipòsit remot :

```
https://github.com/dacomo2122daw2/dacomo2122daw2-html2pdf.git
```

amb l'ordre [git remote](#). L'etiqueta identificadora localment del dipòsit remot serà **html2pdf** en comptes d'**origin**.

3- Descarrega el contingut del dipòsit remot i fusiona'l amb el dipòsit local i el directori de treball local utilitzant l'ordre:

```
git pull <identificador_dipòsit> <nom_branca_remota>
```

a on hauràs de canviar **<identificador_dipòsit>** i **<nom_branca_remota>** pels noms reals dels identificadors del remot i la branca local sincronitzada amb la remota.

4- Fes un **log** per comprovar que HEAD, main i origin/main estan sincronitzats. També un **status** per comprovar que el directori de treball està sincronitzat amb el darrer commit del dipòsit local.

5- Fes un dipòsit local de nom **html2pdf** de tipus **públic** dins del teu compte de **GitHub**. Afegeix l'adreça del dipòsit remot amb **git remote** identificant-lo amb l'alias **h2p**. Verifica que has afegit un nou remot.

6- Puja el contingut del dipòsit local al dipòsit remote identificat amb l'alias **h2p**. Utilitza:

```
git push -u <identificador_dipòsit> <nom_branca_local>
```

a on hauràs de canviar <identificador_dipòsit> i <nom_branca_local> pels noms reals dels identificadors del remote i la branca local sincronitzada amb la remota.

7- Comprova que:

- S'ha actualitzat el dipòsit remot.
- El **log** del dipòsit local demostra que HEAD, main, origin/main i h2p/main estan sincronitzats.

8- Fes les següents accions:

- Modifica LLEGEIX-ME. Esborra l'autor i escriu el teu nom d'usuari de Github. Fes un **add** i un **commit**. El comentari del commit serà "Commit 7 del projecte html2pdf".
- Modifica LLEGEIX-ME. Modifica la data i fes que sigui 13-2-2025. Fes un **add** i un **commit**. El comentari del commit serà "Commit 8 del projecte html2pdf".
- Comprova amb el **log** que HEAD i main no estan sincronitzats amb h2p/main i origin/main.

9- Fes un **push** a **h2p** del dipòsit local per sincronitzar-lo. Comprova que el dipòsit remot es sincronitza i que ara amb un **log** es veu que HEAD i main estan sincronitzats amb h2p/main.

Part 2: Esborrant commits i sincronitzant els dipòsits

1- Esborra el darrer commit actualitzant el directori de treball amb el nou darrer commit. Comprova amb **log** que HEAD i main ja no estan sincronitzats amb h2p/main.

2- **Esborra** el darrer commit també del dipòsit remot utilitzant l'opció **-f** de **git push**.

3- Comprova amb **log** que HEAD i main estan novament sincronitzats amb h2p/main.

Part 3: Sincronitzant branques i fent comprovacions de la localització del HEAD

1- Fes que HEAD passi a apuntar al primer commit. Comprova que HEAD no apunta a main i h2p/main.

2- Crea una nova branca de nom **modificacions**. Passa a la nova branca. Comprova ara a on es troba HEAD i que apunta a l'apuntador **modificacions**.

3- Modifica **html2pdf.php**. A la línia 7 canvia la directiva **require_once** per la directiva **require**. Fes un **add** i un **commit**. El comentari del commit serà "Commit 2 de la branca modificacions del projecte html2pdf". Fes un **log** per comprovar el nou commit local.

4- Fes que HEAD torni a apuntar a main. Fes un **merge** de la branca **main** amb la branca **modificacions**. El missatge del merge serà "Fusió del commit 2 de modificacions amb el commit 7 del main".

5- Fes un **push** de la branca **main** per sincronitzar la branca **main** del dipòsit local amb la branca **main** del dipòsit remot. Comprova que es sincronitzen les branques i amb un **log** comprova que ara HEAD, main i h2p/main estan sincronitzats.

6- Canvia la branca **modificacions**. Fes un **push** de la branca **modificacions** per crear i sincronitzar la branca **modificacions** del dipòsit remot amb la branca **modificacions** del dipòsit local.

7- Comprova que dins de Github s'ha creat una nova branca de nom **modificacions**. Comprova que la branca local i remota estan sincronitzades. Comprova amb un **log** comprova que ara mateix HEAD, modificacions i h2p/modificacions apunten al mateix lloc.

8- Fes que HEAD torni a apuntar a main.

Part 4: Treballant amb fetch

1- Crea una carpeta de projecte de nom **html2pdfalt**. Accedeix a la carpeta i **inicialitza** un dipòsit.

2- Afegeix (i verifica) els remotes per accedir al dipòsit remot :

```
https://github.com/dacomo2122daw2/dacomo2122daw2-html2pdf.git
```

amb l'ordre [git remote](#). L'etiqueta identificadora localment del dipòsit remot serà **html2pdfalt** en comptes d'**origin**.

3- Descarrega el contingut del dipòsit remot sense fusionar-lo amb el dipòsit local i el directori de treball local utilitzant l'ordre:

```
git fetch <identificador_dipòsit> <nom_branca_remota>
```

a on hauràs de canviar **<identificador_dipòsit>** i **<nom_branca_remota>** pels noms reals dels identificadors del remote i la branca local sincronitzada amb la remota.

4- Comprova que si executes:

- **git log** → El resultat és una resposta en blanc perquè localment aquestes baixades encara no han tingut efecte dins de la localització a on es troben les branques locals.
- **ls -ls** → El resultat és una resposta en blanc perquè localment aquestes baixades encara no han tingut efecte dins del directori de treball.
- **git branch -r** → Es mostra **html2pdf/main** => S'ha descarregat la branca main del remote html2pdfalt dins del dipòsit git.
- **git branch** → Es mostra en blanc => Encara no s'ha fet cap commit a cap branca local.

5- Executa ara l'ordre:

```
git merge <identificador_dipòsit>/<nom_branca_remota>
```

a on hauràs de canviar **<identificador_dipòsit>** i **<nom_branca_remota>** pels noms reals dels identificadors del remote i de la branca remota que volem sincronitzar localment.

6- Comprova que si executes:

- **git log** → El resultat és la llista de commits del projecte.
- **ls -ls** → El resultat és la llista de fitxers del projecte que s'han descarregat dins del directori de treball.
- **git branch -r** → Es mostra **html2pdf/main** => S'ha descarregat la branca main del remote html2pdfalt dins del dipòsit git.
- **git branch** → **main** => La branca local main ja té els commits del projecte.

Lliurament de la pràctica

1.- Data Límit: **23-2-24**.

2.- Crea un document amb les respostes (amb captures de pantalla) de nom **daw2_cognom_nom_sm8a2pr5.pdf**.

3.- Envia'm un correu a **cf@collados.org** amb l'assumpte **daw2_cognom_nom_sm8a2pr5** amb:

- El document **daw2_cognom_nom_sm8a2pr5.pdf** adjunt

- Escribeu dins del correu l'adreça **URL** del teu projecte **html2pdf** de **GitHub**.