

Git and GitHub

- What's Git & GitHub & Why U Should Learn.

- ايه هو ال Git وال GitHub 🤖 ال Git هو Distributed Version Control System لو بشتغل في شركة علي project معين ال project ده بيتطور مع الوقت يعني بنضيف فيه new features او بنحل فيه مشاكل او بنعمل enhance لل code المراحل اللي بمر بيها دي اسمها versions . اي program بيكون ليه versions معناه ان كل version مر بمراحل من developments وال Adjustments ال Git بيعمل handle لل versions الخاصه بيا ال Git Is Free and Open Source وميزة انه Open Source دي بيخليني ان ممكن اعمل Contribute لحاجات بداخله .

- ال GitHub عباره عن website بيكون فيه ال source الخاص بال projects الخاصة بيا او بالشركات بيكون فيه ال source الخاص بال libraries اللي بشتغل عليها و projects اللي بتكون open source والناس بيعملوا contribute علي ال projects دي . فيه اكثر من website بيقدمو نفس فكرة ال Github زي GitLab , BitBucket .

ال GitHub بيعمل Simplify ال Using الخاص ب Git يعني بيخليني استخدم ال Git بطريقة بسيطة وببوفر لي امكانيات سلسه .

ممكن استخدم ال Git بدون ال Github يعني الموضوع مش مرتبط بال Github ابدأ انا مش مجبر ان ارفع ال project علي Github انا ممكن ارفعه علي centralized server باستخدام ال Git .

ال Git Has GUI اقدر استخدم ال Git بال Mouse عادي حتي ال Github website ليه Github Program Desktop اقدر استخدمه بال mouse بدون ال Command Line ولكن يفضل استخدام ال Command Line .

ليه Git 🤖 :

- 1 - كل ال Developers اللي شغالين في نفس ال Project بيعملوا Contribute عليه.
- 2 - ممكن اعمل Revert لل Changes واعرف مين عملها واعملها Cancel .
- 3 - ممكن نتعاون علشان نعمل Fix لل Issues ونعمل Create New Features .
- 4 - بيعمل Solve لل Conflicts .
- 5 - بيعمل Organize لل Features .

- ال words اللي هتقابلنا في ال Git & GitHub .

1 - Repository (المستودع) واختصارها كلمة Repo ده اللي بضع فيه ال codes اللي خاصة بال project بتاعي لو عندي 2 project شغال عليهم خاصين بموقع معين ف المفروض يكون عندي 2 repository لكل project .

2 - Branch لو عندي مثلا شركة كبيرة ليها عدة فروع الشركة الكبيره دي هي ال repo والفرع هو ال Branch في ال Git انا باخد Branch من ال project علشان اضيف عليه تعديل بعد كده ابعته تاني .

3 - Local Repo دي الحاجه بتاعتي اللي انا شغال عليها ال PC مثلا .

4 - Remote Repo دي ممكن يكون مثلا ال Github site او ال company server .

5 - Commit (Snapshot or Checkpoint In Your Local Repo) لما بعمل تعديل معين او حلّيت مثلا bug معين بعمل Commit ده بيعمل Checkpoint لل code في ال Local Repo الخاص بيا ولكن بيكون فيه مراحل بعد ال Commit اخري علشان ال code يوصل لل Original Repository .

6 - Clone [From Local Or Remote] ده معناه ان بعمل استنساخ لل Repo اما من ال Local او من ال Remote ال Local يستنسخ منه ولكن في another location طب ليه ممكن اعمل كده 😞 لو عندي project عامله علي IDE وعاوز اجرب اعمله ب framework معين واشوف من هيكون افضل فيهم في ال performance وال security .

7 - Push [Upload Local Changes to Remote] .

8 - Pull [You Pull Changes From Remote Repo To Your Local]

9 - Pull Request [Tell Other About Your Changes To Pull It From Local To Remote] .

- ال Important Notes

1 - لما يكون عندي اكثر من project كل project يكون ليه ال Repository الخاص بيه .

2 - يكون فيه Branch لأي Feature او Enhancement اعمله .

3 - مش محتاج اكون عامل Connect علي Remote Repo وانا شغال انا بكون شغال علي ال Local تمام ولما اخلص شغلي كله اقدر ابعث كل اللي انا عاوزه علي ال Original Repository .

4 - اي حد يقدر يعمل Push and Pull علي حسب ال Permission لان اي Program يشتغل بيه او ال Centralized Server اللي الناس كلها بتدخل عليه بيكون فيه Permission معينه زي مثلا مين يعمل review مين يعمل Connect ومين ميعملش وهكذا .

- Notes.

ال README file امتداده .md اختصار ل Markdown ودي اللغة اللي بكتب بيها او Syntax بيكتب زي text كده ولكن بنسقه بطريقة في ال Github بمعنى لو عاوز اعمل heading معين او font اكبر من font او لو عاوز اعمل مثلا list كل الحاجات دي بيكون ليها specific syntax بيتكتب بيه علشان تحصل.

مهم 👁️ : يفضل استخدام Cmder ع طول بدل cmd لان ال cmd لاتدعم بعض ال commands .

نعمل install ل Cmder 🙌 .

- Create Repo and Clone It.

Create a new repository after registering on GitHub :

- ازاي اعمل clone لل repo 🤔

بعد م اعمل create لل repo من github هاخذ ال link (copy) و هفتح ال Cmder واختار المكان اللي هضع فيه ال projects كلها الخاصه ب github وليكن هيكون ف ال downloads , هعمل create new folder ف ال downloads وهبدأ ان احدهه في ال Cmder بال commands الاتيه :

`cd downloads`

`mkdir My-Github` => Create New Folder

`cd My-Github`

`git clone (repo link)`

ممکن اكتب dir علشان اجيب شوية معلومات عن ال folder .

`cd Git_Course` => Remote Repo Name

- Add And Reset And Commit & Explain Progress.

- بعد م عملت clone لل repo ع ال laptop علشان اشتغل عليه واطبط الشغل وابعتة علشان باقي ال team مثلاً يطلع عليه .

بعد م خلصت عاوز اعمل push للشغل ف ال remote repo (github) .

git status

بيظهر ال status ف ال working directory , بيظهر ايه اللي بيحصل بالظبط .

ف حالتنا هنا لما كتبت ال git status ظهر كما في الصورة الاتيه :

```
Command Prompt
C:\Users\Ayman\Downloads\My-Github\Git_Course>git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
      css/
      index.html

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\Ayman\Downloads\My-Github\Git_Course>
```

شرح الصورة:

- بيقول ان فيه 2 files (Untracked) يعني ميعرفش عنهم حاجه ف لازم اعملهم add لل staging area علشان ابدأ اعملهم commit وبعد كده ارفعهم ع ال remote repo .

ملحوظة 🖐️ : ال css, index.html عاملهم create ف ال project folder و ده للتوضيح فقط .

بمعني اكرر : انظر الصورة الاتيه



- ال working directory ده اللي عامل فيه create لل project folder اللي هو هنا Git_Course بعمل add فيه لكل الشغل بتاعي عن طريق ال git add command لل staging area ودي المنطقه اللي بينطلق منها الشغل النهائي (منطقة التجهيز) بعد كده بعمل git commit علشان اعمل add للي ف ال staging area لل local repo , كده انا لسه ف ال local (لسه ع ال laptop بتاعي) بعد م بخلص كل اللي فات ده بعمل ليه push لل remote repo (github) .

هبدأ انفذ الخطوات :

`git add css\main.css index.html`

`git status`

كتبت `git add` عادي بعد كده `css` ده ال folder name اللي

انا عامله وبداخله عامل فيه `main.css` file بعد كده ال

. `Index.html` file

ممکن اختصر بال command ده `git add *` هيضيف كل ال files ايضاً .

ملحوظة 🖐️ : ال star دي اسمها wild card .

طب لو عاوز اعمل unstage ليهم يعني الغي اضافتهم هستخدم ال command ده :

`git reset head <file name>`

index.html

بعد م عملت add لل files ف ال staging area عاوز اعملها add

ف ال local repo : هستخدم ال command ده :

`git commit -m "any name or message"`

- Push Local Changes To Remote Repository.

بعد م عملت add ف ال staging area عاوز اعمال add ف ال remote repo .

اول حاجه هكتب ال command ده **git branch** علشان اعرف ال all branches اللي موجودة في ال local وال branch اللي شغال عليه ده هيودي ع ال origin اللي هو ال repo اللي موجود ع github .
بعد م يظهر اسم ال branch وهيكون هنا اسمه main .

هكتب ال command ده : **git push origin main**

git push <RemoteName> <BranchName>

origin main

ملحوظة 🖐 : ده ال command اللي بيحيب ال RemoteName : **git remote -v**

اذا كده تم اضافة ال repo ع github .

ملحوظة 🖐 : اي change اعمله ع ال file لازم اعمله **git add <filename>**

- Pull Changes From Remote Repository.

- ازاي اعمل repo ويكون كل واحد ف ال team ممكن انه يشارك فيه 🤔
من ال github هعمل create ل repo عادي وعلشان امكن ال team من الوصول لل project واضافه ال modifications فيه من ال setting الخاص بال repo وهختار Collaborators وهضيف ال members . وكل member هضيفه هيتبع ليه invitation link ع ال gmail .
وال member هيبدأ يعمل الخطوات السابقة علشان يبدأ يعمل modifications ع ال project .
لو فيه file عملته create ع ال remote repo ولكن مش موجود ف ال project ع ال local ازاي بقي اجيب ال file ع ال local 🤔 هستخدم ال command ده :

git pull origin

ال pull بتاخذ ال copy اللي موجود ف ال branch الخاص
بال remote repo تضعه ف ال local وتعمل merge مابين
ال remote repo وال local يعني بيكون فيها git 2 .
(git fetch + git merge)

- Everything About Git Configurations.

- اي program بتعامل معاه بيكون ليه configuration و settings معينه ممكن اعمالها و data خاصه بيا ممكن ان اضيفها ف ال program ف ال git هنضيف ال name, email .
ممكن اغير ال data, setting ع مستوي ال (local) repo او globally .
فيه command بيحيب ال list الخاصه بال available configuration :

`git config --list` Or `git config -l`

علشان اخرج منه اضغط Q .

فيه command بيفتح ال git-config Manual Page ويعرض كل ال configuration :

`git help config`

- ازاي اقدر اعمل edit ع ال configuration دي 🤔 فيه طريقتين عن طريق ال command line او عن طريق ال editor اللي شغال عليه اعدل ال git config .
لو عاوز اجيب specific value وليكن ال user email هستخدم ال command ده :

`git config --global user.email`

ال user.name ده ال configuration اللي عاوز اعرض ال Value الخاصه بيه .

لو مافيش user.email اعمل set لل email عن طريق ال command ده :

`git config --global user.email "اكتب الميل"`

وكذلك مع ال user.name .

فيه command بيعرض كل ال configuration اللي ال git بيجمعها :

`git config -l --show-origin`

لو عاوز احذف حاجه عملتها email مثلا او name هستخدم ال command ده :

`git config --global --unset <الحاجه اللي عاوز احذفها>`

`git config --global --unset user.name`

الطريقه الثانيه لعمل edit بال editor هستخدم ال command ده :

```
git config --global --edit
```

هيفتح ال editor اللي شغال عليه .

لو عاوز اعمل create ل new file ف ال project من ال cmd استخدم ال command ده :

```
touch <اسم الملف>.extension
```

```
touch new_file.txt
```

- Generate And Test Github Public Key.

- ازاي اعمل (Public Key) secure shell key لل github 🤖

ممکن يكون عندي اكثر من public key بدخل بيه ع github من اي مكان ده ببساطه مش بيخليني عند كل action اكتب ال username + password .

ف ال repo بيكون فيه permission فيه specific layer من ال security لازم اللي علوز بدخل ع ال repo يتعمله verify .

ال public key مش خاص فقط ب git & github .

علشان اعمل create ل public key هستخدم ال command ده :

`ssh-keygen -t rsa -b 4096 -C "هنكتب الميـل"`

بعد م اكتب ال command واضغط enter هيعرض ان اعمل password لحماية ال public key وده optional .

كده ال public key اتعمله generate .

هكتب ال command ده علشان يجيب محتوى ال public key وهو بيعرض محتوى اي file بكتبه فيه :

`cat ~/.ssh/id_rsa.pub`

<file اسم ال

ملحوظة 🙌 : نعمل install ل Cmdr .

بعد م يظهر ال content الخاص بال public key هاخدة copy وهروح لل setting ف github واختار SSH and GPG keys ثم New SSH key واضع ال content ف ال key box واعطيه title وممكن اضيف اكثر من key .

علوز بقي اعمل test هيكون عن طريق ال command ده :

`ssh -T git@github.com`

هيطلب ان اكتب ال password الخاص بال public key . وهيظهر ال msg دي

.Hi Ayman717! You've successfully authenticated, but GitHub does not provide shell access

كده بقيت متصل بال github مش محتاج ان ادخل ف كل مرة ال username + password .

- Create Repository From Existing Project.

ازاي يكون فيه project ع ال local واعمله upload ع github 🤖 هروح لل path الخاص بالمكان اللي هنشئ فيه ال repo اي مكان عاوزه .

`cd Documents`

`mkdir repos`

`cd repos`

`git init`

ده بيعمل initialize ل empty repo ف المكان اللي حددته اللي هو Documents .

`touch index.html`

عملت create ل file للتجربة اللي هو index.html عن طريق ال touch command .

`git status`

`git add index.html`

`git commit -m "any name"`

انا كده عندي ال project عاوز اعمله upload ع github هعمل new repo ع ال github بعد كده نختار ال repo link , SSH ف حاله لو عاملين ال Public key لو مش عاملينه هنختار HTTPS .

بعد كده هكتب ال commands دي :

`git remote add origin <repo link>`

`git push -u origin main`

ال -u دي بتخليني اعمل pull بعد كده push .

بعد م اكتب ال command السابق هيطلب مني ال password الخاص بال public key بعد م اكتبه هبدأ تنفيذ ال command .

- Pull Request With Real Examples.

- ال pull request هو request بيخلي ال centralized server يعرف ان عندي new modifications ويعمل request علشان ال admin يعمل accept ويشوف هل مناسبه انه يعمل ليها merge ولا محتاج انه يتعمله maintains و review الاول .
اذا في حاله لو شغال مع team الافضل ان ماعملش push ع طول لازم ال pull الاول علشان مايحصلش . reflects

- هبدا ان اخذ Fork من ال repo من github وال Fork مش هيكون متفعل الا اذا كنت member من organization , بعد كده اختار ال organization اذا كده بقي ال repo في ال repositories list عندي وهيووضح لي انا جاييه من اي organization بعد م اعمل ال modifications الخاصه بي هعمل له commit لل master branch الخاص بي احنا لسه ماروحناش لل organization .

- علوز بقي pull request هضغط ع ال Pull requests Button وهختار New pull request وهيظهر لي شوية معلومات عن اللي هيتم وهيظهر لي ال changes ثم اضغط Create pull request ابدأ ان اعطيه name واكتب any description ثم اضغط Create pull request هيظهر لي شويه معلومات بعد كده هيتبع request لل organization وهيعمل ليهم merge عن طريق الضغط ع Merge pull request ثم Confirm merge ف حاله ان مافيش conflicts .

انا ك member ف ال team ممكن بعد م اخلص ال modifications احذف ال fork واجيبه مرة اخري.
كل ال modifications هتظهر ف ال organization project بعد ال merge process .

- طب لو خدت fork مرة اخري وعملت modifications بس المرة دي هعمل commit ف new branch انا اللي هعمله , ده غير ال branch master , ال new branch ده انا منه هودي ال modification ع ال master branch اللي هو اصلا عندي وبعد م انتهى خالص هاخذ ال master branch وهبعته ع ال remote ك pull request .

طب ليه هعمل Create a new branch 🤔 ف الشركات بيعملوا branch ل اي line of development باخد specific feature انا ك developer وبعملها ف branch وابعتها , بعد كده اعمله . delete

يفضل ان ال pattern الخاص بال branch name يكون بناءً ع ال features اللي بعملها .

بعد م اعمل ال new branch اضغط Propose file change ثم Create pull request , ه يتم عمل Checking ثم اضغط Merge pull request ثم Confirm merge .

بعد كده هروح ع ال Pull request button و هعمل New pull request علشان ده اللي هبعته ع ال remote repo (organization) بعد كده اضغط Create pull request و اكتب title و description ثم اضغط Create pull request .

فيه check Box اسفل ال description موجود ف ال pull request او ال repo settings وهو **Allow edits from maintainers** وده ان ال admin members يقدرُوا يعملُوا maintain ال code قبل ما يعمل merge فلازم اتأكد انها متفعلة .

- Everything About Aliases.

- لما بتعامل مع ال git كثير هحتاج ان اعمل alias لل command اللي بتعامل معاها وال alias ده موجود ف اي system .

ال alias بيفيد جدا ف ان لما يكون عندي command كبير اعمله اسم مختصر استخدمه بيه .

طب ازاي اعمل alias ل command مثلا زي git status 🤖

هكتب ال command ده :

`git config --global alias.st status`

ال st ده الاختصار اللي انا عاوزه لل status وال alias يكون حرفين فأكثر .

Another EX:

`git config --global alias.cm "commit -m"`

cm ده الاختصار اللي عاوزه لل commit -m , استخدمت double quote علشان مايبين ال commit وال -m فيه space ف ال Unix بيعتبرها امرين فلازم استخدم ال double quote او single quote ع حسب ال system اللي شغال عليه .

استخدام ال alias مش بيلغي استخدام ال command الاصلي بمعنى ان ممكن استخدم ال command الاصلي والاختصار اللي انا عملته .

ممكن ال alias اعملها من ال editor اللي شغال عليه عن طريق ان اكتب ال command ده :

`git config --global --edit`

هيفتح ال editor اللي شغال عليه وابدأ اضيف ال alias .

ممكن اعمل search ف google عن git alias list هتظهر list باختصارات عن ال git command .

- Branching And Merging.

- ال branch هو فرع من فروع ال develop سيكون خاص بياك developer علشان تعمل specific task مطلوبه منك او تعمل fix ل specific bug , هاخذ branch هسميه ب proper name وبعد م بخلص الشغل اما بعمل merge لل branch ده مع ال master branch قبل م يتبعك لل remote (github) repo او بيعت ال branch ده ع (github) origin وبطلب pull request علشان ال admin يعمل review لل code .

اول حاجه هكتب ال command ده علشان اشوف كل ال branches اللي موجوده ف ال repo الخاص بيا :

git branch

هكتب ال command ده علشان اعمل create ل new branch :

git branch Scroll

Scroll ده ال branch name .

علشان اروح عليه هكتب ال command ده :

git checkout Scroll

لو عاوز ارجع لل master branch هكتب نفس ال command ولكن بدل Scroll هكتب mater او اسم ال branch اللي عاوز ارجعه .

طب لو عاوز اعمل delete ل branch معين هكتب ال command ده :

git branch -d Scroll

ممکن ف ال delete ايضا استخدم ال command ده :

git branch -D Scroll

- الفرق بين ال command الاول والثاني ان -d بتعمل save delete يعني بيخلي ال git يعمل check هل ال branch فيه modification متعملهاش merge ولا لا لو لا مش هيحذفه . لكن ال -D بيعمل delete حتي لو فيه modifications .

فيه طريقه اخري لعمل branch وبروحله ع طول عن طريق ال command ده :

`git checkout -b Dev-Feature`

ال Dev-Feature ده ال branch name .

طب ازاي اعمل rename لل branch 🤖 عن طريق ال command ده :

`git branch -m development`

ال development ده ال new name لل branch .

بعد كده هبدأ ان اعمل create لل files علشان الشغل بتاعي .

بعد م اخلص هعمل merge لل branch بتاعي مع ال master branch وابتعت ال master ع ال origin عن طريق ال commands الاتيه . (الطريقه الاولى)

`git checkout master`

`git merge Development`

`git branch -d Development` => Optional

`git push origin master`

(الطريقة الثانية)

هبعث ع ال origin ع طول مع ال pull request من غير م اعمل merge .

`git checkout Development`

`git push origin Development`

`git branch -d Development` => Optional

- بعد كده من github ال admin هيدخل ع ال repo هيلقي ان فيه pull request هيضغط ع Compare & pull request وهيكتب title, description ثم هيضغط Create pull request ثم Merge pull request ثم Confirm merge .

- Mastering Stash Part 1.

- ال stash command ده بقدر من خلاله اعمل hiding يعني لو عملت create ل files وفيها modifications او files مش هضيف فيها modifications حاليا ومش عاوز ارفعها ف ال branch حاليا , هستخدم ال stash هي زي Box بخزن فيها ال files .
ف بعد م اعمل create لل file وعاوز اخزنهم اكتب ال command ده :

git stash

ممکن اتأكد ان ال files اتعملها stash عن طريق ال command ده :

git stash list

وده بيعرض كل ال stash اللي اتعملها create .

فيه command بيظهر لي كل ال files ف ال project وهو ls

ملحوظة 🖐️ : بعد م اعمل modifications ف file معين ف ال project وعاوز ارفعه ف ال project ع github لازم الاول اجيب نسخه من ال latest repo اللي على github عندي قبل م اعمله push وهيكون عن طريق ال command ده :

git pull origin master

بعد كده اعمل push عن طريق

git push origin master

طب بعد م عملت حاجات تانيه ورفعتها عاوز بقي ارجع ال files اللي عملتها stash 🤔 هستخدم ال command ده :

git stash pop

بعد كده اعمل ليها commit وارفعها ع github .

ملحوظة 🖐️ : لو عاوز اعمل file واكتب فيه هستخدم ال command ده :

echo "Ayman Sedik" > about_me.txt

about_me ده ال file name وال txt ده ال extension.

- Mastering Stash Part 2.

- لو عملت create ل file وعملته stash وكملت شغل , بعد كده عملت another file وعملته stash كل file بيكون ف box مختلف عن الاخر وكل file بيكون ليه id مختلف عن الاخر .

عاوز اضيف file ف ال stash واعطيله name هستخدم ال command ده :

`git stash save "Ayman Text"`

معني ال command ان عندي file ضفته ف ال stash وسميته Ayman Text .

طيب لما بعمل `git stash pop` علشان اجيب ال files هجيب اخر box ف ال index اتعمله create وهيحفز ال box . طب لو عاوز اجيب box من غير م يتحفز ؟ هستخدم ال command ده :

`git stash apply`

ال apply بياخد ال file اللي عامل فيه modifications وهيبييه ف مكانه ف ال stash (box) ومش هيحفز ال box . ده الفرق بين ال apply وال pop .

لما بستخدم pop, apply بيجيب اخر file انا عملته stash طب ازاي اجيب اي file معين 🤔 هيكون بال id الخاص بيه ف هستخدم ال command ده :

`git stash pop stash@{id}`

Or

`git stash apply stash@{id}`

لو عاوز اعمل delete ل stash بال files اللي بداخله استخدم ال command ده :

`git stash drop`

ده هيحفز اخر stash .

طب لو عاوز احذف stash معين بال command ده :

`git stash drop stash@{id}`

فيه command بيعرض كل اللي ف ال stash وهو :

`git stash show`

ده هيعرض اخر stash .

طب لو عاوز اعرض stash معين بال command ده :

`git stash show stash@{id}`

طب لو عاوز احذف كل ال stashes اللي موجوده بال command ده :

`git stash clear`

- Restore And Clean.

- لو عملت file معين وعملته add لل staging area وعاوز اعمله Cancel add من ال staging area علشان يكون Untracked وعاوز ايضا اعمله delete . طب ازاي 🤔 اعمل كده

ازاي اعمل unstage لل file بمعنى انه يكون untracked هيكون عن طريق ال command ده :

`git restore --staged <اسم الملف>`

لو عملت git status هلاحظ انه بقي Untracked .

لو عندي اكثر من file عاوز اعمله restore استخدم ال command ده :

`git restore --staged *`

طب لو عاوز احذف ال files 🤔 بعد م اعمل ليهم restore بال command السابق, هعمل show الاول علشان اتأكد ان دي ال files اللي عاوز احذفها وهيكون عن طريق ال command ده :

`git clean -n`

لو file مش عاوز احذفه اضيفه بال `git add` عادي بعد كده الباقي احذفه بال command ده :

`git clean -f`

- Resetting The Head.

- ال reset بتمكني ان احذف ال commits اللي مش محتاجها او اللي بيكون فيها مشاكل .
اولا علشان اقدر احذف ال commit لازم اكون عارف ال head وده ف ال git هو pointer بيعمل point على اخر commit معمول .
فيه command بيعرفني اخر commit انا عملته ال head بتاعه خاص ب ايه 🤔 وهو :

git log

- طيب علشان احذف commit لازم اخلي ال head بتاعه يشاور ع اللي بعده , كل commit بعمله بيكون ليه hash بيظهر لما بعمل git log وبيكون موجود ايضا ف ال commit part ع github وده ب اختصار رقم بيكون بجانب ال commit بيتم اعطاءه لل commit علشان تسريع عملية ال search
اذا علشان احذف commit معين ف محتاج ال hash الخاص بال commit اللي بعده علشان اخليه هو ال head وايضا علشان اخليه هو اخر commit موجود واللي بعده كله يتحذف .
هكتب ال command ده :

git reset --hard <hash number>

- ده بيحذف ال commit بالكامل بال changes , وال commit اللي بعد اللي اتحذف بيكون هو ال head وبيعمل point ع اللي بعده اللي هو اخر commit .
بعد كده اعمل update لل remote repo عن طريق ال command ده :

git push origin master -force

- طب لو عاوز احذف مجموعه من ال commit هيكون 🤔 ازاي هخلي ال head الخاص باول commit عاوز احذفه يشاور ع ال commit اللي قبل اللي عاوز احذفه ف مجموعه ال commits وده هيكون بالطريقة الاتيه .

git log

علشان اخرج من ال log اضغط Q

بعد كده هجيب ال hash الخاص بال commit اللي قبل اللي عاوز احذفه ف مجموعه ال commits واضعه ف ال command ده :

git reset --hard <hash num>

git push origin master --force

- Ignoring Files And Directories.

لو عاوزين نعمل ignore ل files معينه علشان متترفعش لل project ع github يكون ازاي 🤔
وليه هكون محتاج ان اعمل كده 🤔 احيانا بيكون فيه files مش مهم ان ارفعها ف ال project بسبب
مثلا حجمها زي node_modules او بسبب عدم اهميتها .

ازاي اقدر اعمل كده 🤔

هعمل create ل file هسميه .gitignore. عن طريق ال command ده :

touch .gitignore

ده بيخلي ال git يعمل ignore ل files معينه .

لو عملت show لل files اللي عندي بال ls command مش هيظهر ولكن هيظهر لما اعمل git status
هعمل ع ال file ده edit من ال editor وفتحه ف ال editor عن طريق ال command ده :

code .gitignore

بداخل ال file لو عاوز مثلا اعمل ignore ل كل ال files اللي ال extension الخاص بيه log ما يعملش
ليه upload اعملها ازاي 🤔 هكتب ف ال file كده *.log. اذا كده اي file ال extension الخاص بيه
log مش هيعمل ليه upload وهكذا مع اي file عاوز اعمله ignore .

طب لو عاوز اعمل ignore لكل ال files اللي ال extension الخاص بيه log. ماعدا file معين اعملها
ازاي 🤔 هكتب ف ال .gitignore file ع ال editor كده !vip.log .
ال vip ده ال file اللي مش عاوز اعمله ignore .

طب لو عملت folder ومش عاوز اعمله include بمعنى انه ميكونش ف ال folder الرئيسي الخاص بال
project يكون ازاي 🤔

هعمل create ل folder للتجربة هيكون بال command ده :

mkdir node_packs

ال node_packs ده اسم ال folder الي ال nodejs بيعمله create ف ال project وده اللي مش عاوز
اعمله include .

`cd node_packs`

كده انا داخل ال folder وبعمل create ل file .

`touch package1.js`

علشان اخرج من ال folder اكتب ال command ده :

`cd ..`

`git status`

ومن `git status` هأخد اسم ال folder (directory) اللي هو `node_packs` وهضعه ف ال `.gitignore` file ع ال editor .

ممکن اكتب اسم file من ال project ف ال `.gitignore` file. عاوز اعمله ignore بدون اي مشاكل .

طب لو عملت ignore ل file وعاوز اعمله push حتي وهو معمول له ignore يكون ازاي 🤔

هيكون عن طريق ال force ف هستخدم ال command ده :

`git add -f <اسم الملف>`

اخيرا هكتب ال command ده :

`git add *`

بعد كده اعمل commit لل files وارفعها .

ملحوظة 🖐 :

Search about git ignore patterns on google.

- Tagging And Releasing Part 1.

- ال tag ده بيمكني ان اعمل mark ع جزء معين من ال project ف فترة زمنية معينة , بيتم استخدامهما لل versions ف ال project .

اي project بيكون ليه version وكل version بيكون احدث من اللي قبله .
من مميزات ال tag ان كل version ممكن اعمله download لوحدة .

فيه 2 types من ال tag :

الاول lightweight tag .

الثاني Annotated tag .

- ال lightweight tag : هي tag مرتبطه بال commit وبتاخذ ال messages بتاعتها من ال mesaages اللي كتبته ف ال commits , اعملها ازاي 🤖 بال commands الاتيه :

`git tag`

ده علشان اظهر انا عندي كام tag

`git tag v1.0`

ال v1.0 ده اسم ال tag .

`git push origin v1.0`

ف github هنلاحظ انه عمل add ل tag 1 .

- ال Annotated tag : اعملها ازاي 🤖

`git tag -a v2.0 -m "any message"`

ال -a معناها ان هستخدم ال Annotated tag

`git push origin v2.0`

ملحوظة 🙌 : اي modifications هعملها هتظهر ف كل ال versions الا لو كل version ليه branch.

- Tagging And Releasing Part 2.

لو فيه مجموعة tags كتير جدا ف اقدر اعمل list لل tags اللي محتاجها عن طريق ال command ده :

"*. اسم الاصدار اللي عاوزه" -l git tag

"v2.*" -l git tag

ده هيجيب كل ال versions الثانية فقط .

github فيه Release بيكون فيه كل action حصل ف ال version بال commits .

ممکن نعمل create ل release خاص بال tag , من github اضغط Create release واعطيه title و اكتب describe .

فيه check box وهو ال pre-release وده لو عاوز مثلا اوضح حاجه قبل م اعمل upload لل version , لو مش عاوز اضغط ع طول Publish release . فيه ميزة ال compare ان اعمل مقارنه بين ال versions وبعضها .

فيه ايضا ميزة Draft a new release بحيث ان اعمل release ولكن مش هعمل له launch حاليا .

علشان اعمل delete لل tag هستخدم ال command ده :

v1.0 -d git tag

ده بيحذف ال tag ولكن locally .

لو عاوز احذف ال tag من ال remote repo هستخدم ال command ده :

v1.0 --delete git push origin

- Summary of the most important commands.

***** Upload Project On GitHub *****

```
git init

git config --global user.name "Your Username On GitHub"
git config --global user.email "Your Gmail"

git config --global --unset <الحاجه اللي عاوز احذفها> => For Deleting
git config --global --unset user.name

git remote add origin <Repo Link>

git add .

git commit -m "Any Message"

git log => to check that commit is ok (Optional)

git push origin master
```

***** Upload After Editing *****

```
git add .

git commit -m "Any Message"

git log

git push origin master
```

***** If the project admin make a collaborate for me and I want to get the latest version of the project to make my edit *****

Create New Folder On Computer

```
git clone <Project Repo Link>
```

// Upload After My Editing

```
git add .

git commit -m "Any Message"

git log

git push origin master
```

***** ممكن من جيت هب اعمل برانش جديد وارفع عليه تعديلاتي وفي الاخر الادمن يدمج كل البرانشز *****

git branch => علشان اعرف ال branches اللي موجوده

git checkout -b <New Branch Name> => يجيب ال new branch اللي هضيف فيه التعديلات الخاصه بي

***** لو انا من احد اعضاء التيم وعاوز اجيب عندي اخر نسخه للبروجيكت من جيت هب علشان اعدل فيه يكون اراي *****

من ملف البروجيكت ع اللاب

```
git pull origin master

git log
```

***** مسحت حاجه بدون قصد من البروجيكت ومعملتش commit اجيبها اراي *****

```
git reser --hard => ده هيحيب اخر commit
```

- ازاي نعمل host لل Website ع GitHub pages 🤔

How to host a Website in GitHub pages??

```
git init
```

```
git remote add origin <repo link>
```

```
git status <optional, to check>
```

```
git add .
```

```
git commit -m "any name or message"
```

```
git log <Optional, to verify that the commit was created>
```

```
git push origin master
```

* من ال setting الخاص بال repo اختار Pages واختار ال Branch الي فيه ال files علشان اعملها hosting واعمل save وانتظر لحد ميعمل ال Built ثم اعمل refresh .