

Git & Github

Partie 1

Fait par Joel Sandé
Hiver 2021

Survol : nous allons parler un peu de tout ça

- Repository / dépôt
- Commit
- Branch
- Merge
- Fork
- Pull Request

Ce qu'il fait savoir pour débuter

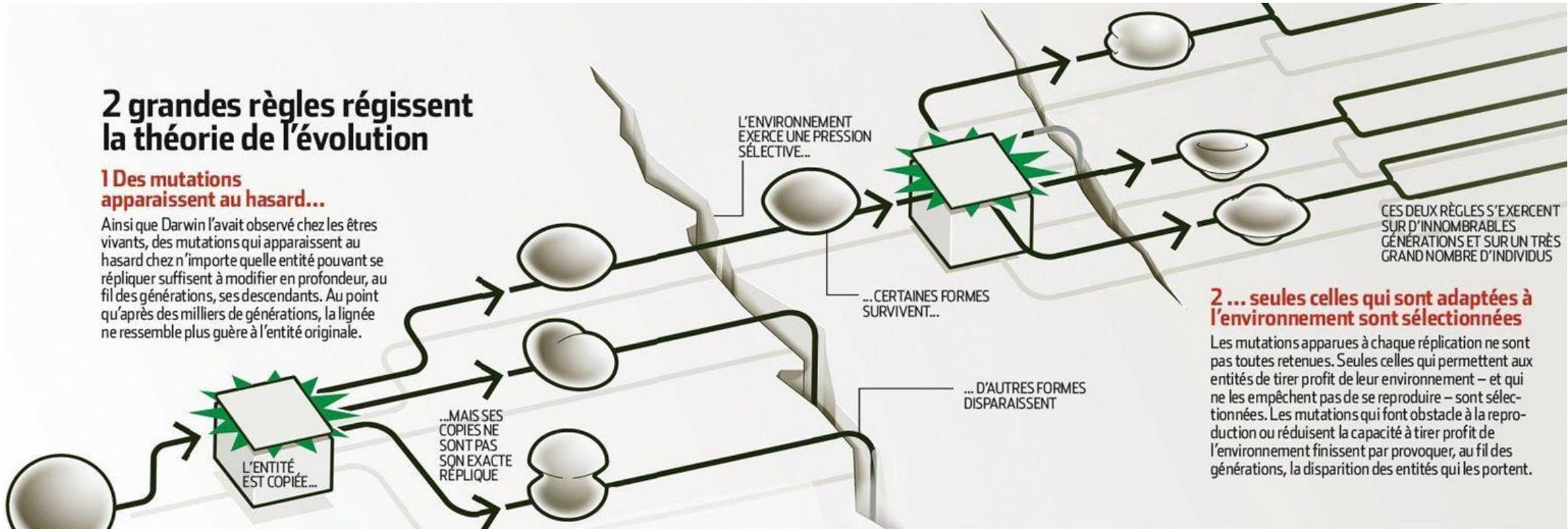
- Sachez que :
- Vous n'avez pas besoin de connaitre de langage de programmation particulier pour travailler avec Git ou Github.
- Un écrivain peut utiliser Git et Github, ne serait-ce que pour de la collaboration. Pas besoin d'avoir de lien avec la programmation.

Mise en situation

- Je commencez un travail, et je pense à plein de choses en même temps. Dans ce travail, à la page 5, je dessine un dinosaure; un dinosaure que je dois colorier, décider s'il faut lui donner des bras ou des ailes, ou des bras et des ailes, je lui donne des poils, des plumes ou une peau imperméable, .. Jusque là, **avec le Drive**, je dois faire un choix linéaire: Je le colorie en vert, je lui donne des pattes et des ailes, je lui fais cracher du feu, ...je suis en train de décrire un « dragon » en éliminant toutes les autres possibilités. Bonnes nouvelle : **avec Git et Github**, je peux me permettre d'avoir un dragon qui crache le feu, et un qui n'en crache pas, un dragon bleu et un vert, un avec un bec, et un avec des cros, .. Et même avant de décider que ce serait un dragon, c'était d'abord un reptile dans une version plus ancienne; je peux donc à partir d'une version plus ancienne, décrire un crocodile, et dans une version beaucoup plus ancienne, décrire une baleine. Cette dernière baleine, dans des versions plus récentes, pourra être soit bleue, ou soit brune, sachant que je peux produire les 2 copies : **Avec Git, je peux écrire plusieurs histoires.**

Analogie avec la théorie de l'évolution

- C'est un peu comme la théorie de l'évolution :



<https://www.science-et-vie.com/archives/l-irresistible-extension-de-la-theorie-de-l-evolution-37187>

Git

- Voyez-vous, ces différentes versions du projet, Git vous permet de les sauvegarder de façon intelligente et structurée : **Contrôle de version**.
- En d'autres mots, il vous fait un historique du projet.
- Il le contrôle bien, même en situation de **travail collaboratif**.
- En somme, Git est un système qui permet de faire un contrôle intelligent et structuré de versions d'un projet, même en circonstance de travail collaboratif (**même si plusieurs personnes travaillent en même temps sur le même dossier**).

Git vs Github

- Git : Logiciel de contrôle de version
- Inventé sous Linux, il a été créé par l'équipe de Linux Dorval, père de Linux.
- Sachez qu'il est possible de faire du Git (versionnage de projets) sans passer par Github et inversement.
- Git peut être local
- Github : Siteweb, service web ou tu peux te login et avoir un compte, pour faire du Git.
- Sachez qu'il est possible d'être sur le site Github sans faire du Git.
- Github est obligatoirement en ligne
- Github vient d'être acheté par Microsoft (si l'information est encore à jour)

Nomenclature

Le nom GitHub est composé du mot « git » faisant référence à un système de contrôle de version open-source et le mot « hub » faisant référence au réseau social bâti autour du système Git, mais aussi à une plate-forme de correspondance qui est appelée en anglais un « hub ».

<https://fr.wikipedia.org/wiki/GitHub>

Allons s'enregistrer et se login sur Github

- Rendez-vous sur le site : www.github.com
- Sign-up et login.
- Voici un guide d'utilisation de Github que nous allons décortiquer au fur et à mesure : <https://guides.github.com/activities/hello-world/>

Create a new repository / Crée un nouveau dépôt



Search or jump to...



Pull requests Issues Marketplace Explore

Add a README file peut être pratique

Je n'ai pas cliqué Add .gitignore, mais un peu plus tard en local on verra ce que c'est.

ET enfin, cliquez sur Create repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Owner *



Repository name *

Animal_Prehistorique



Great repository names are short and memorable. Need inspiration? How about scaling-lamp?

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.

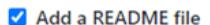


Private

You choose who can see and commit to this repository.

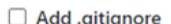
Initialize this repository with:

Skip this step if you're importing an existing repository.



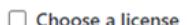
Add a README file

This is where you can write a long description for your project. [Learn more](#).



Add .gitignore

Choose which files not to track from a list of templates. [Learn more](#).



Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

This will set main as the default branch. Change the default name in your [settings](#).

Create repository

Pas d'espace dans le nom, sinon GitHub va vous imposer une barre-d'union.

Vous pouvez choisir la visibilité public (visible par tout le monde) ou privée (visible par vous seul).

Voici enfin notre premier dépôt 😊

The screenshot shows a GitHub repository page. At the top, there's a dark header with the GitHub logo, a search bar, and navigation links for Pull requests, Issues, Marketplace, and Explore. Below the header, the repository name 'joel225 / Animal_Prehistorique' is displayed, followed by a '(Private)' badge. A navigation bar below the repository name includes links for Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. The main content area shows a summary of the repository: 'main' branch, 1 branch, 0 tags. It lists a single commit: 'joel225 Initial commit' (13bb4ce, now, 1 commit). This commit contains a file named 'README.md' with the content 'Initial commit'. To the left of the repository details, a blue callout box with a white border contains the text: 'Vous vous souvenez que j'ai coché l'option Add a README file : et bien il est là.' A blue arrow points from the bottom right corner of this box towards the 'README.md' file in the repository list. On the right side of the page, there are sections for About (with a note: 'No description, website, or topics provided.'), Readme (with a link to 'Readme'), Releases (with a note: 'No releases published' and 'Create a new release'), and Packages (with a note: 'No packages published' and 'Publish your first package'). At the very bottom of the page, there's a footer with links for © 2021 GitHub, Inc., Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, and Blog.

Vous vous souvenez
que j'ai coché l'option
Add a README file :
et bien il est là.



Search or jump to...



Pull requests Issues Marketplace Explore

joel225 / Animal_Prehistorique Private

Unwatch ▾

Code

Issues

Pull requests

Actions

Projects

Security

Insights

Settings

main ▾

1 branch

0 tags

joel225 Initial commit

README.md

Initial commit

README.md

Animal_Prehistorique

Jouez avec ces 3 boutons : vous pouvez :

- Allez sur un des fichiers de votre dépôt.
- Créer un nouveau fichier
- Télécharger un fichier.

Go to file

Add file ▾

Code ▾

About

Go to file

Create new file

Upload files

1 commit

2 minutes ago

No description, website, or topics provided.

Readme

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

© 2021 GitHub, Inc.

Terms

Privacy

Security

Status

Docs



Contact GitHub

Pricing

API

Training

Blog

About

Je me crée un nouveau fichier

A screenshot of a GitHub repository page for 'joel225 / Animal_Prehistorique'. The repository is private. The 'Code' tab is selected. A new file named 'Mammifere.txt' is being created in the 'main' branch. The file content is currently empty, indicated by the number '1' at the top left. A blue arrow points from the text 'Je vais l'appeler Mammifere.txt.' to the file name 'Mammifere.txt' in the navigation bar. A red note in a box states: 'NB : L'extension peut être un .txt, docx, .java, cpp, ... Comme je l'ai dit d'entrée, Github, ce n'est pas que de la prog. Un écrivain peut utiliser Github.'

Search or jump to... Pull requests Issues Marketplace Explore

joel225 / Animal_Prehistorique Private

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Security Insights Settings

Animal_Prehistorique / Mammifere.txt in main Cancel changes

Edit new file Preview Spaces 2 No wrap

1

Je vais l'appeler Mammifere.txt.

NB : L'extension peut être un .txt, docx, .java, cpp, ...
Comme je l'ai dit d'entrée, Github, ce n'est pas que de la prog. Un écrivain peut utiliser Github.

<> Edit new file

Preview

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.  
2  
3 Exemple, nous pouvons citer :  
4  
5 1) Les vertébrés, parmi lesquels :  
6 Amphibiens  
7 Crocodiliens  
8 Mammifères  
9 Oiseaux  
10 Poissons  
11 Serpents et lézards  
12 Tortues  
13  
14 2) Mollusques  
15 Bivalves  
16 Céphalopodes  
17 Gastéropodes  
18  
19 3) Arthropodes  
20 Arachnides  
21 Crustacés  
22 Insectes  
23 Myriapode  
24  
25 4) Annélide  
26  
27 5) Cnidaire  
28  
29 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```

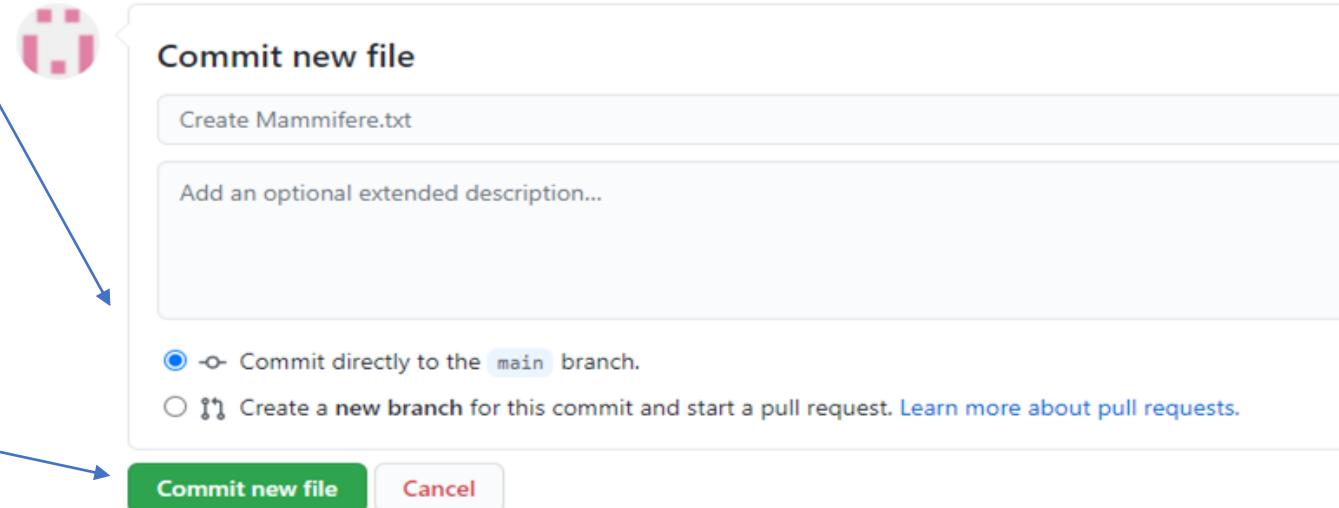
Commit dans la branche principale.

Nous allons rester dans la branche principale pour l'instant.

Nous verrons par la suite en quoi consiste la création d'une nouvelle branche.

N'oubliez pas de cliquer sur **Commit** avant de quitter.
Commit = Save

```
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
19 3) Arthropodes
20 Arachnides
21 Crustacés
22 Insectes
23 Myriapode
24
25 4) Annélide
26
27 5) Cnidaire
28
29 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```



C'est fait !
Il est sauvé !

À présent je peux cliquer sur [mammifere.txt](#) ou [Go to file](#) puis sur [Édit this file](#) pour continuer à l'éditer.

The screenshot shows a GitHub repository named "joel225 / Animal_Prehistorique". The "Code" tab is selected. At the top, it shows "main" branch, 1 branch, 0 tags, and a commit from "joel225" titled "Create Mammifere.txt" made 22 minutes ago. Below the commit, there are two files: "Mammifere.txt" and "README.md". To the right, there are buttons for "Go to file", "Add file", and "Code". On the far right, there are sections for "About" (no description), "Readme", "Releases" (no releases), and "Packages" (no packages). A blue arrow points from the text "C'est fait !" to the "Code" button, and another blue arrow points from the text "Il est sauvé !" to the "Initial commit" message in the commit list.

joel225 / Animal_Prehistorique Private

Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

joel225 Create Mammifere.txt 114556f 22 minutes ago 2 commits

Mammifere.txt 22 minutes ago

README.md Initial commit 1 hour ago

README.md

Animal_Prehistorique

About

No description, website provided.

Readme

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

```
14 2) Mollusques  
15 Bivalves  
16 Céphalopodes  
17 Gastéropodes  
18  
19 3) Arthropodes  
20 Arachnides  
21 Crustacés  
22 Insectes  
23 Myriapopes  
24  
25 4) Annélide  
26  
27 5) Cnidaire  
28  
29 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp  
30
```

Commit changes = Sauvegarder les changements.

Il contient 2 parties :

1) Le titre que je donne au premier changement apporté. Il faut que le titre soit assez court (moins de 50 caractères).

2) Un résumé (facultatif du titre) si on a besoin d'apporter des précisions sur le titre.



Commit changes

Je revois l'orthographe de Myriapope

Je revois l'orthographe de Myriapopes (parmi les Arthropodes) en le mettant au pluriel plutôt qu'au singulier.

-o- Commit directly to the `main` branch.

⌂ Create a new branch for this commit and start a pull request. [Learn more about pull requests](#).

Commit changes

Cancel

```
11 Serpents et lézards
12 Tortues
13
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
19 3) Arthropodes
20 Crustacés
21 Insectes
22 Myriapodes
23
24 4) Annélide
25
26 5) Cnidaire
27
28 Source provenant de https://www.teteamodeler.com/écologie/biologie/vivant/classification-animal.asp
29
```



Commit changes

Je supprime les Arachnides (Arthropodes)

Je supprime les Arachnides, de la famille des Arthropodes. À présent, j'ai 3 espèces d'Arthropodes plutôt que 4 que j'avais auparavant.

→ Commit directly to the `main` branch.

↗ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes

Cancel

joel225 / Animal_Prehistorique Private

Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code About

joel225 Je supprime les Arachnides (Arthropodes) ... 0ce9e80 3 hours ago 4 commits

Mammifere.txt Je supprime les Arachnides (Arthropodes) 3 hours ago

README.md Initial commit

Je supprime les Arachnides (Arthropodes)

Je supprime les Arachnides, de la famille des Arthropodes. À présent, j'ai 3 espèces d'Arthropodes plutôt que 4 que j'avais auparavant.

No releases published Create a new release

Packages

No packages published Publish your first package

je supprime les Arachnides (Arthropodes)

Initial commit

Je supprime les Arachnides (Arthropodes)

Je supprime les Arachnides, de la famille des Arthropodes. À présent, j'ai 3 espèces d'Arthropodes plutôt que 4 que j'avais auparavant.

No releases published Create a new release

Packages

No packages published Publish your first package

Je retourne à présent sur la dernière mise-à jour (commit) effectuée.

© 2021 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

Je supprime les Arachnides (Arthropodes)

[Browse files](#)

Je supprime les Arachnides, de la famille des Arthropodes. À présent, j'ai 3 espèces d'Arthropodes plutôt que 4 que j'avais auparavant.

main

 joel225 committed 3 hours ago Verified

1 parent b192d2d commit 0ce9e8017e9f068db362828990d13fbe0fdf0dbf

 Showing 1 changed file with 0 additions and 1 deletion.

Unified Split

1 Mammifere.txt

@@ -17,7 +17,6 @@ Céphalopodes

17 17 Gastéropodes

18 18

19 19 3) Arthropodes

20 - Arachnides

21 20 Crustacés

22 21 Insectes

23 22 Myriapodes

Github me présente en rouge le dernier élément supprimé. Un ajout aurait été représenté en vert.

Remarque : Ce fichier (commit) parent a un ID : il... restera unique dans tout votre projet; vous pourrez à tout moment, même dans 2 semaines, après avoir beaucoup avancé dans votre travail, revenir le récupérer, et recommencer à travailler dessus si vous voulez développer un autre travail sous un angle complètement différent.

0 comments on commit 0ce9e80

 Lock conversation



Write

Preview

Leave a comment

H B I E <> O E E @ ↗ ↙ ↖ ↖

À présent cliquez sur `<>` code en haut à gauche, et [Go to file](#). Cliquez sur le nom du fichier.

Attach files by dragging & dropping, selecting or pasting them.



[Comment on this commit](#)

Private

Actions Projects Security Insights Settings

main Animal_Prehistorique / Mammifere.txt

joel225 Je supprime les Arachnides (Arthropodes) ...

1 contributor

28 lines (21 sloc) | 512 Bytes

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées
2
3 Exemple, nous pouvons citer :
4
5 1) Les vertébrés, parmi lesquels :
6 Amphibiens
7 Crocodiliens
8 Mammifères
9 Oiseaux
10 Poissons
11 Serpents et lézards
12 Tortues
13
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
```

Latest commit 0ce9e80 hours ago History

Raw Blame

History for Animal_Prehistorique / Mammifere.txt

Commits on Apr 16, 2021

Je supprime les Arachnides (Arthropodes) ...
joel225 committed 4 hours ago
Verified 0ce9e80

Je revois l'orthographe de Myriapode ...
joel225 committed 4 hours ago
Verified b192d2d

Create Mammifere.txt
joel225 committed 5 hours ago
Verified 114556f

Newer Older

Avec Historique, j'ai la liste de toutes les versions de mon fichier.

History for Animal_Prehistorique / Mammifere.txt

-o Commits on Apr 16, 2021

- Je supprime les Arachnides (Arthropodes) ...
joel225 committed 4 hours ago
- Je revois l'ortographe de Myriapope ...
joel225 committed 4 hours ago
- Create Mammifere.txt
joel225 committed 5 hours ago

Newer Older

Je clique sur une des versions antérieures, et je peux voir ce qui a été retiré (rouge) et ce qui a été ajouté (vert)

Je revois l'ortographe de Myriapope

Je revois l'orthographe de Myriapopes (parmi les Arthropodes) en le mettant au pluriel plutôt qu'au singulier.

main

joel225 committed 4 hours ago Verified

1 parent 114556f commit b192d2d2be9e1f84c1ba5ac

Showing 1 changed file with 1 addition and 1 deletion.

Mammifere.txt

...	@@ -20,7 +20,7 @@ Gastéropodes
20 20	Arachnides
21 21	Crustacés
22 22	Insectes
23	- Myriapope
23	+ Myriapopes
24 24	4) Annélide
25 25	
26 26	
....	

0 comments on commit b192d2d

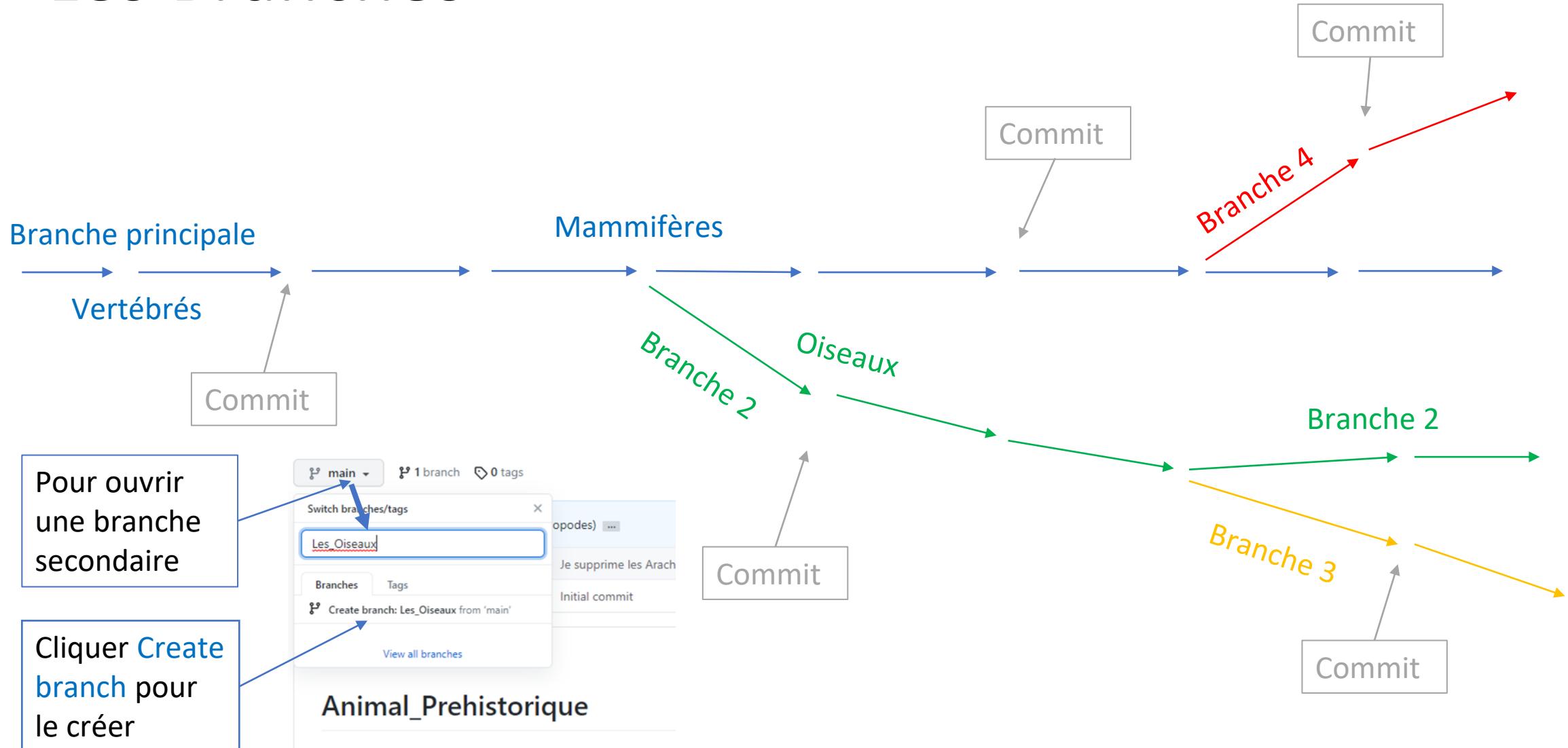
Write Preview

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Comment

Les Branches



main ▾ 2 branches 0 tags

Switch branches/tags

Find or create a branch...

Branches Tags

✓ main default

Les_Oiseaux

View all branches

witch branches or tags

Animal_Prehistorique

À présent, on peut voir qu'on a 2 branches :

- La principale
- Les_Oiseaux

opodes) ... 0ce9e80 5 hours ago 4 commits

Je supprime les Arachnides (Arthropodes) 5 hours ago

Initial commit 7 hours ago

Go to file Add file ▾ Code ▾

About

No description, website, or topics provided.

Readme

Releases

No releases published Create a new release

Packages

Les_Oiseaux ▾ 2 branches 0 tags

This branch is even with main.

joel225 Je supprime les Arachnides (Arthropodes) ... 0ce9e80 5 hours ago 4 commits

Mammifere.txt Je supprime les Arachnides (Arthropodes) 5 hours ago

README.md Initial commit 7 hours ago

Go to file Add file ▾ Code ▾

About

No description, website, or topics provided.

Readme

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

README.md

Animal_Prehistorique

Graphs(Insights) and Network

- Il faut passer à la version Pro pour voir ce que ça donne en graphique.

main -> Animal_Prehistorique / Mammifere.txt

joel225 Update Mammifere.txt

Latest commit 4bad71c 18 minutes ago History

1 contributor

34 lines (24 sloc) | 835 Bytes

Raw Blame

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.
2
3 Exemple, nous pouvons citer :
4
5 1) Les vertébrés, parmi lesquels :
6 Amphibiens
7 Crocodiliens
8 Mammifères
9 Oiseaux
10 Poissons
11 Serpents et lézards
12 Tortues
13
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
19 3) Arthropodes
20 Crustacés
21 Insectes
22 Myriapodes
23
24 4) Annélide
25
26 5) Cnidaire
27
28 Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large, celui du règne. \n
29 Le règne animal inclue l'homme. \n
30 Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par tous les scientifiques. \n
31
32
33
34 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```

Branche principale

Remarquez bien le paragraphe en bas. Ça va nous servir à comprendre le merge.

Les_Oiseaux

Animal_Prehistorique / Mammifere.txt

Go to file

...

joel225 Update Mammifere.txt

Latest commit sb6937f 12 minutes ago History

1 contributor

31 lines (23 sloc) | 735 Bytes

Raw Blame

1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.

2

3 Exemple, nous pouvons citer :

4

5 1) Les vertébrés, parmi lesquels :

6 Amphibiens

7 Crocodiliens

8 Mammifères

9 Oiseaux

10 Poissons

11 Serpents et lézards

12 Tortues

13

14 2) Mollusques

15 Bivalves

16 Céphalopodes

17 Gastéropodes

18

19 3) Arthropodes

20 Crustacés

21 Insectes

22 Myriapodes

23

24 4) Annélide

25

26 5) Cnidaire

27

28 La classification du règne animal présentée ici est la classification courante retenue notamment pour la majorité des manuels scolaires.
29 Nous allons parler du règne des Oiseaux dans les quelques lignes qui suivent.

30

31 Source provenant de <https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp>

Branche 2 = Les_Oiseaux

Remarquez bien le paragraphe du bas. Ça va nous servir à comprendre le merge.

Pull request et Merge

En gros si on remarque que ce qu'on a dit dans la branche 2 n'est pas mal du tout, on peut le fusionner avec la branche principale. Je demande à ma branche principale d'accepter d'incorporer mon idée.

Diagram illustrating the Pull Request and Merge process:

- A horizontal timeline shows commits moving from left to right.
- A green arrow points from the last commit on the secondary branch (labeled 'B') up to the main branch (labeled 'A'), indicating a pull request.
- Another green arrow points from the merged state back down to branch B, indicating the merge.

Screenshot of a GitHub repository interface:

- Repository Summary:** Les_Oiseaux had recent pushes 32 minutes ago. 2 branches, 0 tags.
- Status:** This branch is 3 commits ahead, 3 commits behind main.
- Commit History:**
 - joel225 Update Mammifere.txt (commit 5b6937f, 32 minutes ago, 7 commits)
 - Mammifere.txt (Update Mammifere.txt, 32 minutes ago)
 - README.md (Initial commit, 23 hours ago)
- Actions:** Go to file, Add file, Code, Pull request, Compare.
- Branches:** Compare & pull request (button highlighted with a blue oval).
- Details:** About, No description, website, or topics provided. Readme, Releases (No releases published, Create a new release), Packages (No packages published, Publish your first package).

Text boxes:

- Top Left:** En gros si on remarque que ce qu'on a dit dans la branche 2 n'est pas mal du tout, on peut le fusionner avec la branche principale. Je demande à ma branche principale d'accepter d'incorporer mon idée.
- Bottom Right:** Pour cela, je me positionne sur (le dernier commit de) la branche 2, et je fais un Pull-request sur la branche main.

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).



base: main ▾



compare: Les_Oiseaux ▾



Can't automatically merge. Don't worry, you can still create the pull request.



J'ai ajouté des détails sur les manuels scolaires

Write

Preview



J'ai mergé la branche des oiseaux contenant les détails sur les manuels scolaires.

Attach files by dragging & dropping, selecting or pasting them.



Create pull request



Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

J'ai ajouté des détails sur les manuels scolaires #1

Open joel225 wants to merge 3 commits into `main` from `Les_Oiseaux`

Conversation 0 Commits 3 Checks 0 Files changed 1

joel225 commented 1 minute ago

J'ai mergé la branche des oiseaux contenant les détails sur les manuels scolaires.

joel225 added 3 commits 1 hour ago

- o Classification retenue par les manuels scolaires a69454c
- o Parlons des oiseaux d7b7567
- o Update Mammifere.txt 5b6937f

Add more commits by pushing to the `Les_oiseaux` branch on `joel225/Animal_Prehistorique`.

This branch has conflicts that must be resolved
Use the [web editor](#) or the [command line](#) to resolve conflicts.

Conflicting files
`Mammifere.txt`

Resolve conflicts

Merge pull request You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

S'il y a des conflits signalés, il faut les corriger en déplaçant les textes. Mais la vraie chose qu'on fait ici, c'est qu'on supprime quelques lignes.

<<<<<
=====
>>>>>
du coté de la branche qui fait la requête.

1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.
2
3 Exemple, nous pouvons citer :
4
5 1) Les vertébrés, parmi lesquels :
6 Amphibiens
7 Crocodiliens
8 Mammifères
9 Oiseaux
10 Poissons
11 Serpents et lézards
12 Tortues
13
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
19 3) Arthropodes
20 Crustacés
21 Insectes
22 Myriapodes
23
24 4) Annélide
25
26 5) Cnidaire
27
28 Les_Oiseaux
29 La classification du règne animal présentée ici est la classification courante retenue notamment pour la majorité des manuels scolaires.\n30 Nous allons parler du règne des Oiseaux dans les quelques lignes qui suivent.
31
32 Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large, celui du règne. \n33 Le règne animal inclue l'homme. \n34 Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par tous les scientifiques. \n35
36 Source provenant de <https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp>

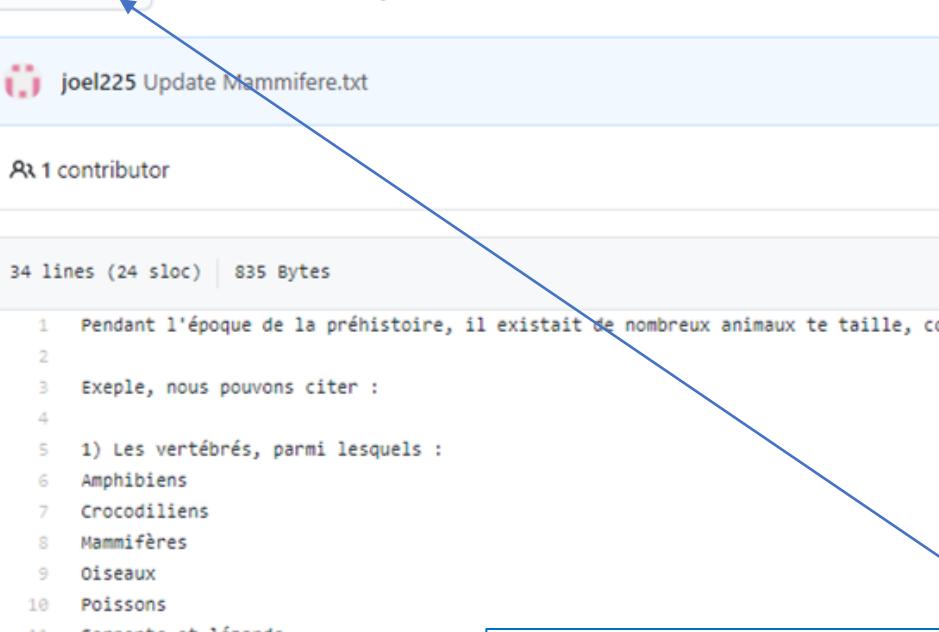
Réarrangement des morceaux
pour régler les conflits.

Détails sur les manuels scolaires #1

ix and main and committing changes → Les_Oiseaux

	Mammifere.txt	1 conflict	Prev ^	Next v	⚙️	Mark as resolved
1	Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.					
2						
3	Exemple, nous pouvons citer :					
4						
5	1) Les vertébrés, parmi lesquels :					
6	Amphibiens					
7	Crocodiliens					
8	Mammifères					
9	Oiseaux					
10	Poissons					
11	Serpents et lézards					
12	Tortues					
13						
14	2) Mollusques					
15	Bivalves					
16	Céphalopodes					
17	Gastéropodes					
18						
19	3) Arthropodes					
20	Crustacés					
21	Insectes					
22	Myriapodes					
23						
24	4) Annélide					
25						
26	5) Cnidaire					
27						
28	Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large, celui du règne. \n					
29	Le règne animal inclue l'homme. \n					
30	Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par tous les scientifiques. \n					
31						
32	Les_Oiseaux					
33	La classification du règne animal présentée ici est la classification courante retenue notamment pour la majorité des manuels scolaires.\n					
34	Nous allons parler du règne des oiseaux dans les quelques lignes qui suivent!					
35						
36	Source : http://www.education.gouv.fr/medias/documents/2012/06/05/classification_animal.pdf					

Mark as resolved / Marquer comme résolu
puis Commit pour sauver la modification.

A red arrow originates from the 'main' dropdown menu at the top left and points towards the 'joel225' commit entry in the list.

joel225 Update Mammifere.txt

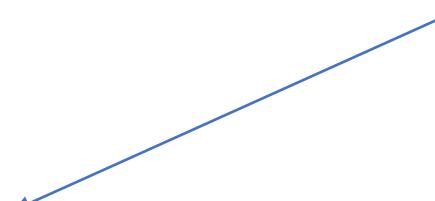
Latest commit 4bad71c 1 hour ago 

1 contributor

34 lines (24 sloc) | 835 Bytes

Raw Blame   

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.  
2  
3 Exemple, nous pouvons citer :  
4  
5 1) Les vertébrés, parmi lesquels :  
6 Amphibiens  
7 Crocodiliens  
8 Mammifères  
9 Oiseaux  
10 Poissons  
11 Serpents et lézards  
12 Tortues  
13  
14 2) Mollusques  
15 Bivalves  
16 Céphalopodes  
17 Gastéropodes  
18  
19 3) Arthropodes  
20 Crustacés  
21 Insectes  
22 Myriapodes  
23  
24 4) Annélide  
25  
26 5) Cnidaire  
27  
28 Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large, celui du règne. \n  
29 Le règne animal inclue l'homme. \n  
30 Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par tous les scientifiques. \n  
31  
32  
33  
34 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```



À présent, observez le fichier de la branche principale. Le merge de sa part, bien qu'a été REQUESTED par la branche 2 (Les_Oiseaux), n'a pas encore été accepté par la branche principale.

 joel225 Merge branch 'main' into Les_OiseauxLatest commit scaf164 16 minutes ago 

1 contributor

36 lines (27 sloc) | 1.04 KB

Raw Blame   

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.  
2  
3 Exemple, nous pouvons citer :  
4  
5 1) Les vertébrés, parmi lesquels :  
6 Amphibiens  
7 Crocodiliens  
8 Mammifères  
9 Oiseaux  
10 Poissons  
11 Serpents et lézards  
12 Tortues  
13  
14 2) Mollusques  
15 Bivalves  
16 Céphalopodes  
17 Gastéropodes  
18  
19 3) Arthropodes  
20 Crustacés  
21 Insectes  
22 Myriapodes  
23  
24 4) Annélide  
25  
26 5) Cnidaire  
27  
28 Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large, celui du règne. \n  
29 Le règne animal inclue l'homme. \n  
30 Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par tous les scientifiques. \n  
31  
32 Les_Oiseaux  
33 La classification du règne animal présentée ici est la classification courante retenue notamment pour la majorité des manuels scolaires.\n  
34 Nous allons parler du règne des Oiseaux dans les quelques lignes qui suivent.  
35  
36 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```

Ensuite, observez le fichier [Mammifere.txt](#) de la branche 2 (Les_Oiseaux).

Vous constatez que c'est cette branche 2 que je règle les conflits avant de la proposer à la principale.

J'ai ajouté des détails sur les manuels scolaires #1

Edit Open with ▾

Merged

joel225 merged 4 commits into main from Les_Oiseaux 1 minute ago

Conversation 0

Commits 4

Checks 0

Files changed 1

+3 -1

Je termine mon Pull-request avec un Merge.

Donc en pratique, ma branche `Les_oiseaux` peut-être supprimée à cet instant, car les 2 branches ont la même information.



joel225 commented 13 days ago

J'ai mergé la branche des oiseaux contenant les détails sur les manuels scolaires.



...

Reviewers

No reviews



Assignees

No one—assign yourself



Labels

None yet



Projects

None yet



Milestone

No milestone



Linked issues

Successfully merging this pull request may close these issues.



None yet

Notifications

Customize

Unsubscribe

You're receiving notifications because you're watching this repository.



Pull request successfully merged and closed

You're all set—the `Les_Oiseaux` branch can be safely deleted.

Delete branch



Write

Preview

H

B

I

E

<

>

≡

≡

☒

@

✉

↶

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Comment

Lock conversation

main Animal_Prehistorique / Mammifere.txt

joel225 Merge branch 'main' into Les_Oiseaux

1 contributor

36 lines (27 sloc) | 1.04 KB

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses
2
3 Exemple, nous pouvons citer :
4
5 1) Les vertébrés, parmi lesquels :
6 Amphibiens
7 Crocodiliens
8 Mammifères
9 Oiseaux
10 Poissons
11 Serpents et lézards
12 Tortues
13
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
19 3) Arthropodes
20 Crustacés
21 Insectes
22 Myriapodes
23
24 4) Annélide
25
26 5) Cnidaire
27
28 Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large,
29 Le règne animal inclue l'homme. \n
30 Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par
31
32 Les_Oiseaux
33 La classification du règne animal présentée ici est la classification courante retenue notamment pour la majorité de
34 Nous allons parler du règne des Oiseaux dans les quelques lignes qui suivent.
35
36 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```

Latest commit 5caf

Les_Oiseaux Animal_Prehistorique / Mammifere.txt

joel225 Merge branch 'main' into Les_Oiseaux

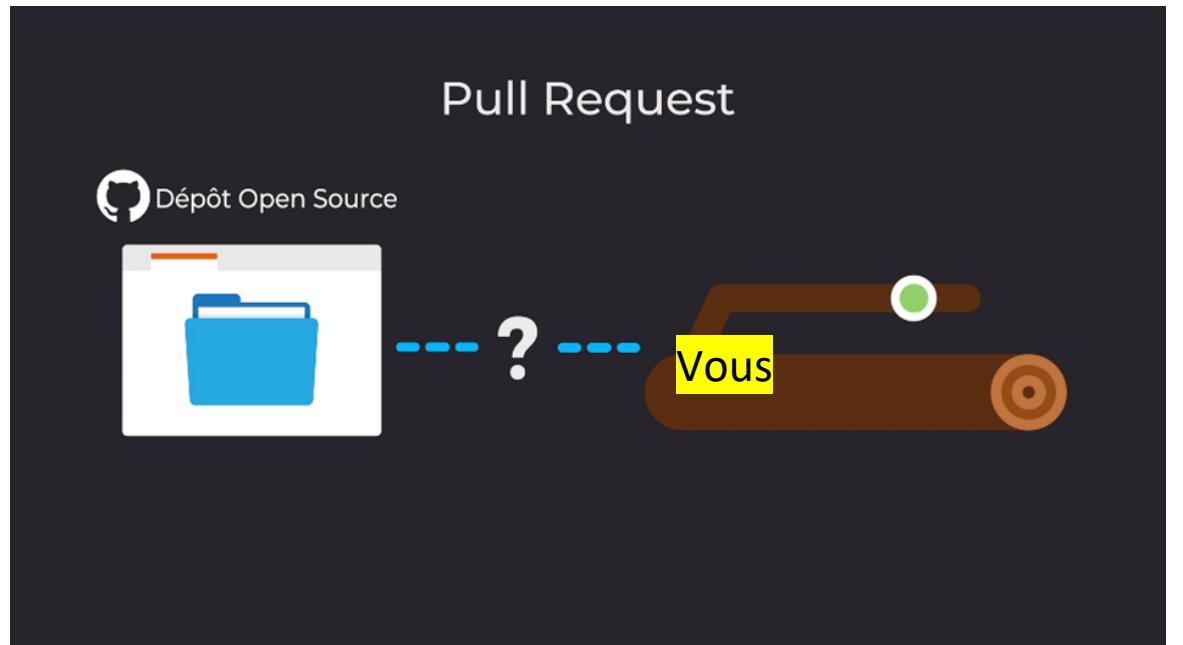
Latest commit 5caf

1 contributor

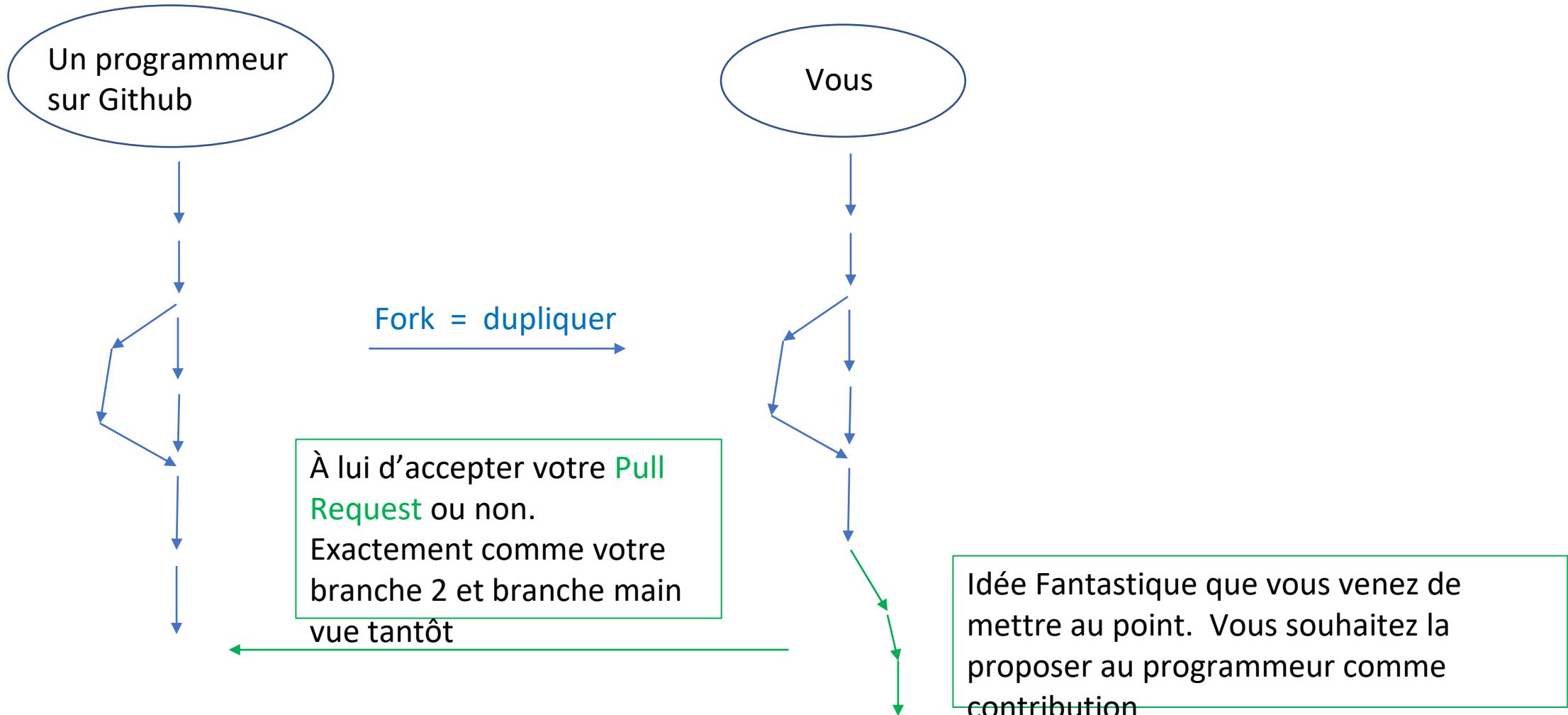
36 lines (27 sloc) | 1.04 KB

```
1 Pendant l'époque de la préhistoire, il existait de nombreux animaux de taille, couleurs, morphologies aussi diverses que variées.
2
3 Exemple, nous pouvons citer :
4
5 1) Les vertébrés, parmi lesquels :
6 Amphibiens
7 Crocodiliens
8 Mammifères
9 Oiseaux
10 Poissons
11 Serpents et lézards
12 Tortues
13
14 2) Mollusques
15 Bivalves
16 Céphalopodes
17 Gastéropodes
18
19 3) Arthropodes
20 Crustacés
21 Insectes
22 Myriapodes
23
24 4) Annélide
25
26 5) Cnidaire
27
28 Ceci est la fiche de synthèse sur l'organisation du règne animal. Le mot animal est à prendre au sens le plus large, celui du règne. \n
29 Le règne animal inclue l'homme. \n
30 Si certains réfutent cette classification pour des raisons religieuses, c'est la classification qui est retenue par tous les scientifiques. \n
31
32 Les_Oiseaux
33 La classification du règne animal présentée ici est la classification courante retenue notamment pour la majorité des manuels scolaires.\n
34 Nous allons parler du règne des Oiseaux dans les quelques lignes qui suivent.
35
36 Source provenant de https://www.teteamodeler.com/ecologie/biologie/vivant/classification-animal.asp
```

- Même principe avec une open source qui n'est pas la vôtre.

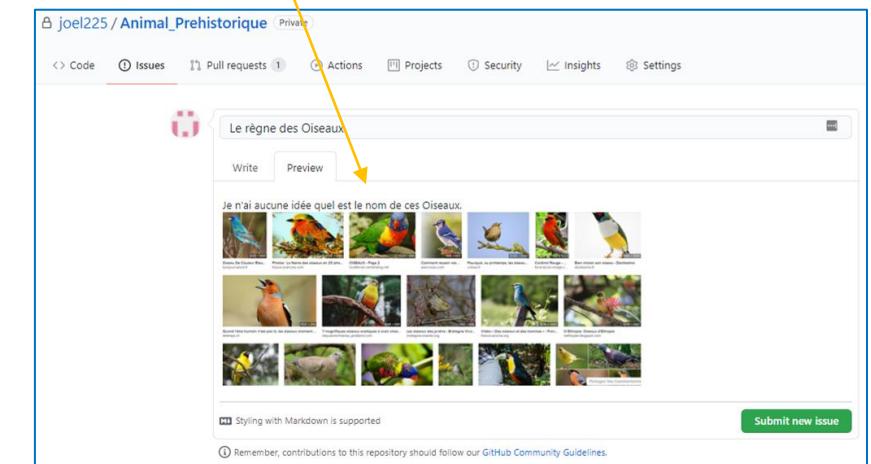
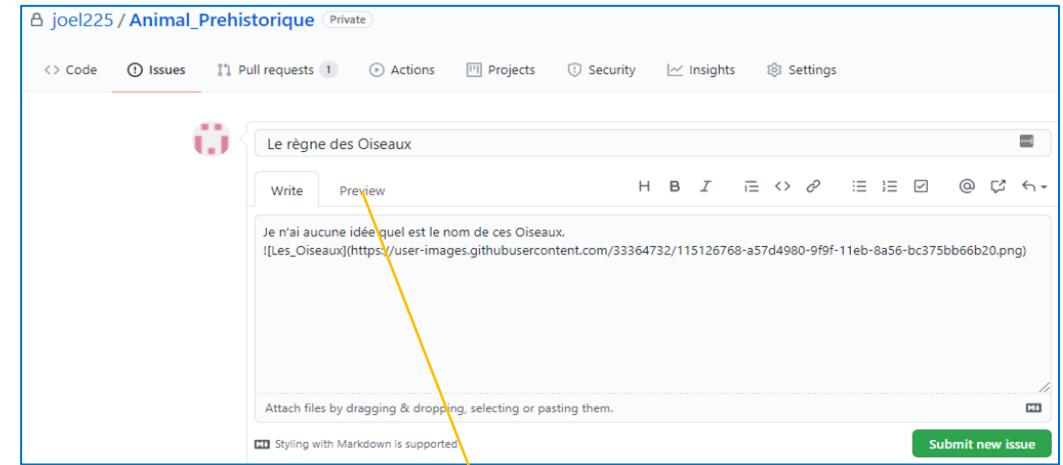


Fork a repository et Pull Request



La rubrique : Issues

- « Issues » c'est un endroit où laisser un commentaire à propos d'un projet. Il est généralement relié aux Bogues.
- On peut aussi utiliser « Issues » pour poser des questions.
- Vous pouvez ajouter une image avec un print-screen.
- Vous pouvez cliquer sur preview pour voir l'image.



Le règne des Oiseaux

Write

Preview



Je n'ai aucune idée quel est le nom de ces Oiseaux.

![Les_Oiseaux](https://user-images.githubusercontent.com/33364732/115126768-a57d4980-9f9f-11eb-8a56-bc375bb66b20.png)

[Voici la référence de mon image](https://duckduckgo.com/?q=oiseaux&iax=images&ia=images)

Attach files by dragging & dropping, selecting or pasting them.

Styling with Markdown is supported

Remember, contributions to this repository should follow our GitHub Com

Vous pouvez ajouter un lien hypertexte pour la référence, ... en suivant ce formatage

Toutes les informations du style vous sont données ici

Le règne des Oiseaux

Write Preview

Je n'ai aucune idée quel est le nom de ces Oiseaux.

Voici la référence de mon image

Styling with Markdown is supported

Submit new issue

Le règne des Oiseaux #2

[Edit](#)[New issue](#)

joel225 opened this issue 1 minute ago · 1 comment



joel225 commented 1 minute ago

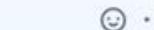
Je n'ai aucune idée quel est le nom de ces Oiseaux.



[Voici la référence de mon image](#)



joel225 commented now



...



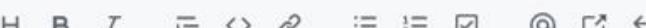
...

C'est juste un test. pas besoin d'y répondre.



Write

Preview



Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Close issue

Comment

Assignees

No one—assign yourself



Labels

None yet



Projects

None yet



Milestone

No milestone



Linked pull requests

Successfully merging a pull request may close this issue.

None yet



Notifications

Customize

Unsubscribe

You're receiving notifications because you're watching this repository.

1 participant



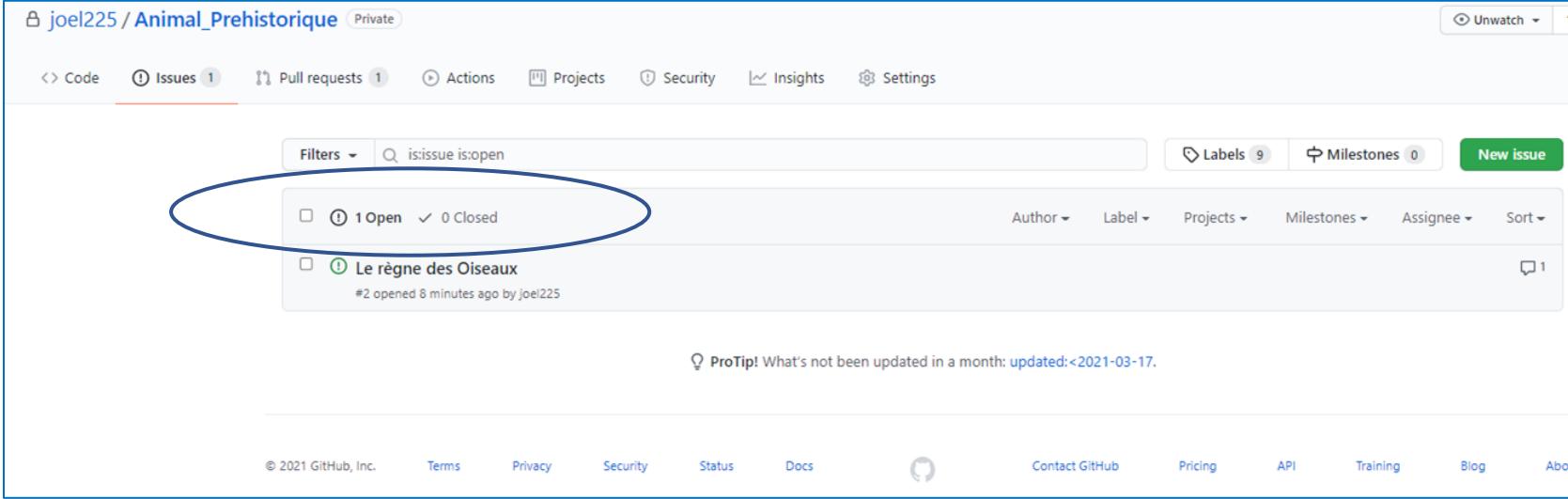
Lock conversation

Pin issue

Transfer issue

Delete issue

Vous pouvez y mettre un libellé, définir le projet lié à ce problème, ... en jouant avec les options à droite.



The screenshot shows a GitHub repository page for 'joel225 / Animal_Prehistorique'. The 'Issues' tab is selected, showing one open issue. A blue oval highlights the filter bar at the top, which displays 'is:issue is:open' and a summary of '1 Open' and '0 Closed' issues. Below the filter bar, the specific issue is listed: '#2 Le règne des Oiseaux' opened 8 minutes ago by joel225. The GitHub interface includes standard navigation links like Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings, along with filters, labels, milestones, and a 'New issue' button.

Voici la liste des différents problèmes soumis.

Un problème est ouvert jusqu'à ce qu'il soit fermé. Quand il est résolu, il est fermé.

Un visiteur peut soumettre un problème sur n'importe quel projet.

C'est le propriétaire du projet, ou celui qui a ouvert le problème qui peut fermer le problème.

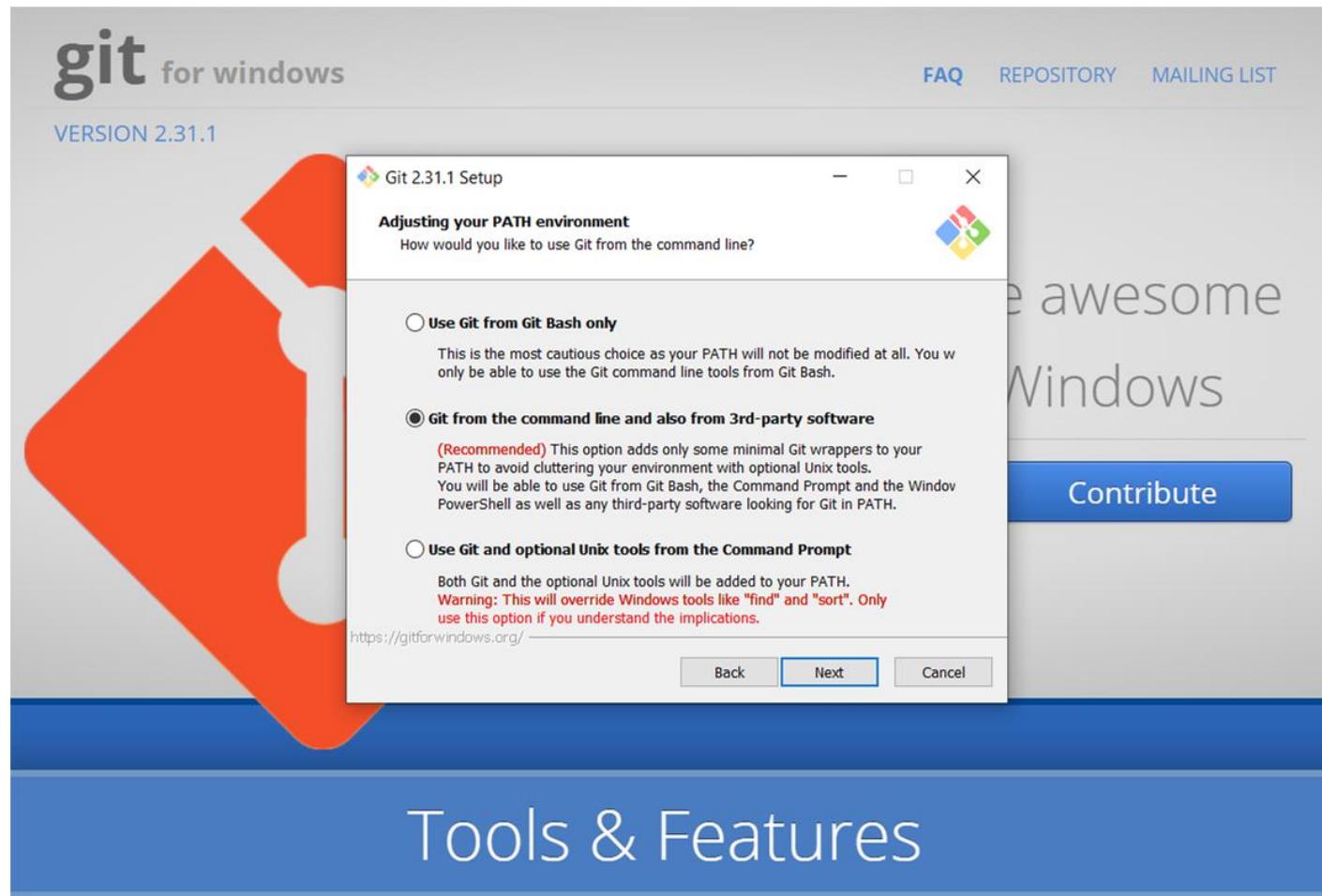
Comment installer Git sur son ordinateur

Comment installer Git sur son ordinateur

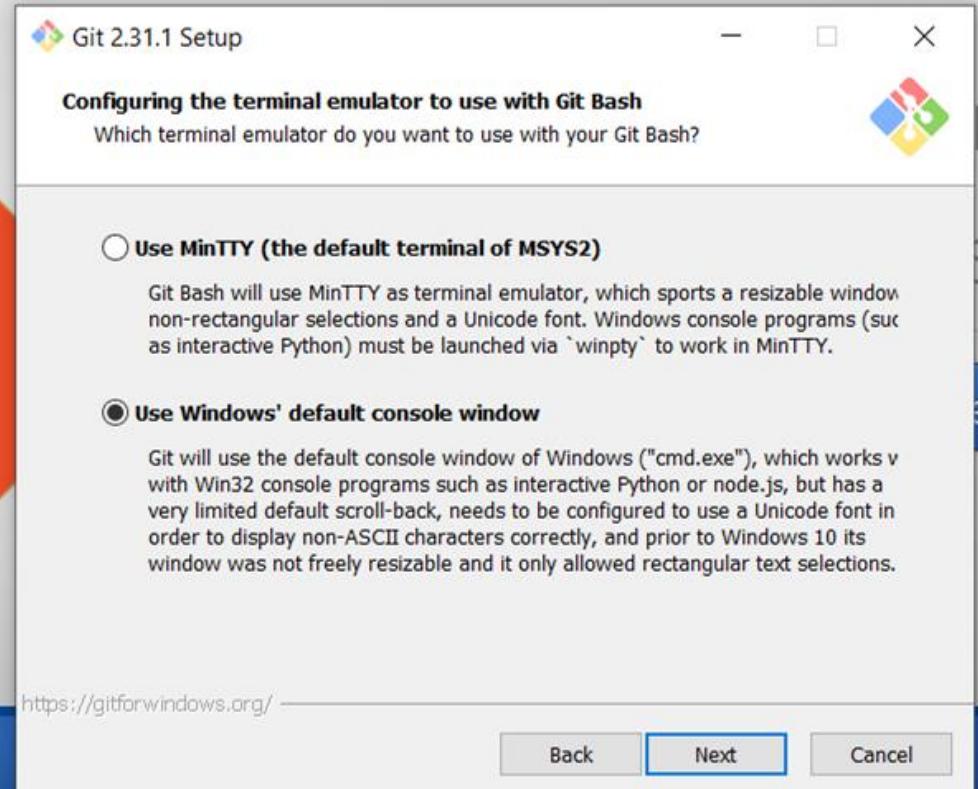
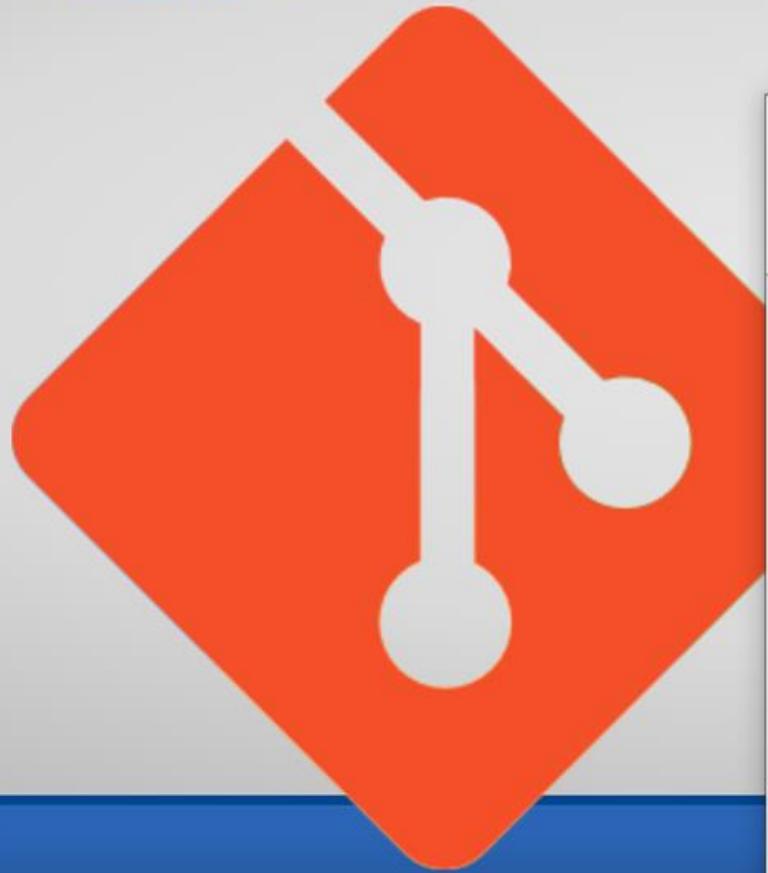
- Voici où télécharger Git pour Windows OS <https://gitforwindows.org/>
- ou encore <http://www.git-scm.com/>
- Sur Mac OS : tapez la command `git version` puis install
- Si cette méthode ne fonctionne pas, Aller le téléchargez ici
<http://www.git-scm.com/download/mac>

Une façon simple de procéder est de cliquer sur [next](#) à toutes les étapes sans changer de paramètres.

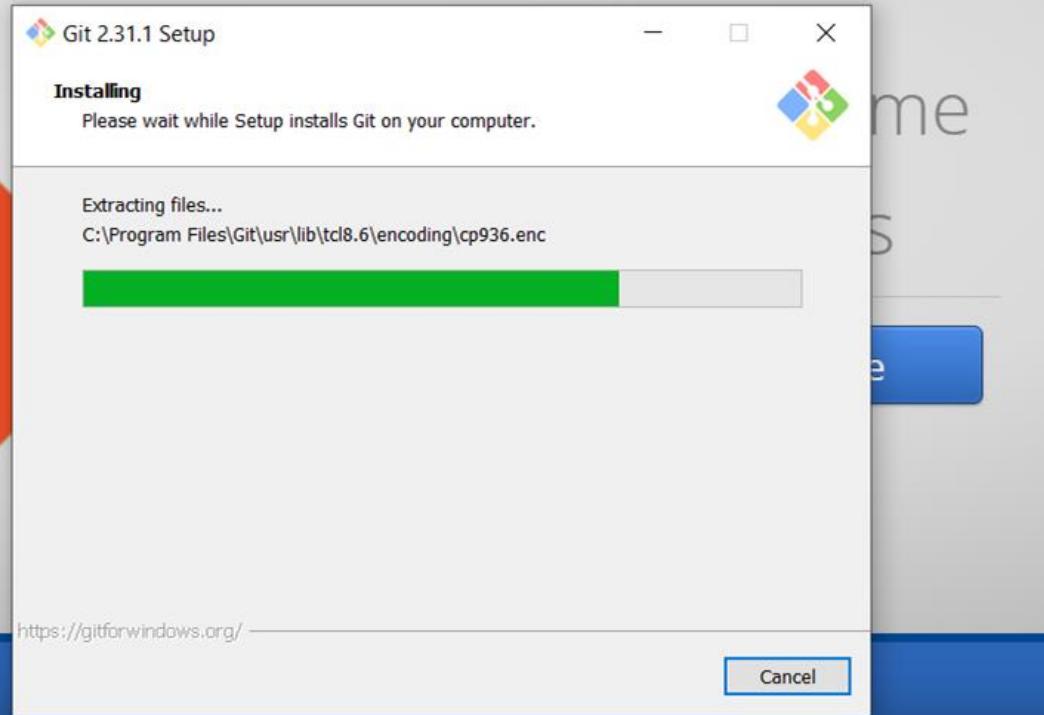
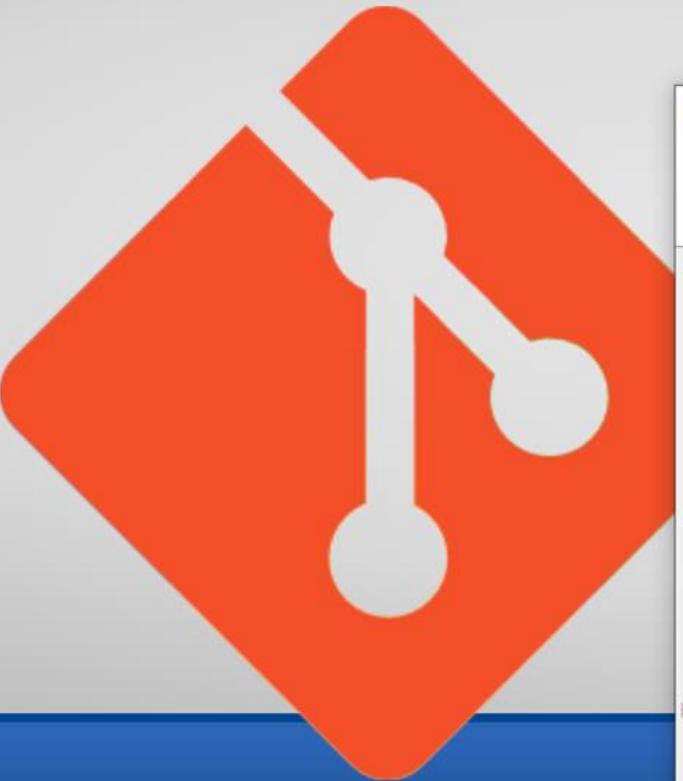
OU au pire, sélectionnez les 2 étapes spécifiques que je vous montre. C'est comme vous voulez.



VERSION 2.31.1



Tools & Features



Tools & Features

Une fois installé, ouvrons Git Bash. Vous le trouverez aussi via la barre de recherche

Voici un audit des différents changements qui ont été apportés d'une année à l'autre.

<file:///C:/Program%20Files/Git/ReleaseNotes.html>



Les Lignes de commandes

Les lignes de commande

- Il existe un passé, un passé lointain, très lointain, où l'informatique consistait en des lignes de commandes. Aaaayoyyy ! Et oui, ce n'était pas accessible à tout le monde...
- À cette époque, il n'y avait pas d'interface graphique. Créer un fichier, supprimer un fichier, écrire dans un fichier, chercher un dossier, changer de direction, afficher le contenu d'un fichier, ... , c'était l'ère des lignes de commandes.
- À la base, un ordinateur c'est ça : un **invité de commandes**, c'est **le terminal de votre ordinateur**. Les grandes startup ont par la suite modélisé en graphique pour le rendre accessibles à tous par de simples cliques.
- Avec Git, on va surtout utiliser l'invité de commandes.

- Une des façons de travailler avec Git est de taper des commandes. Ceci vous permettra de travailler localement sans passer par le site Github.
- Voici une liste de commandes incontournables :
 - `cd` : Change directory / changer de répertoire
 - `pwd` : Print working directory
 - `ls` : Liste all item present in this directory
 - `clear` : effacer tout, très pratique comme commande
 - `cat nom_du_fichier` : permet de voir tout le contenu d'un fichier
 - `echo "Hello word" >> nom_du_fichier` me permet d'écrire dans un fichier

Git : Faisons un peu d'exploration via les lignes de commandes

```
MINGW64:/c/Users/jsande
jsande@jsande-p MINGW64 ~
$ -



MINGW64:/c/users/jsande
jsande@jsande-p MINGW64 ~
$ pwd
/c/Users/jsande

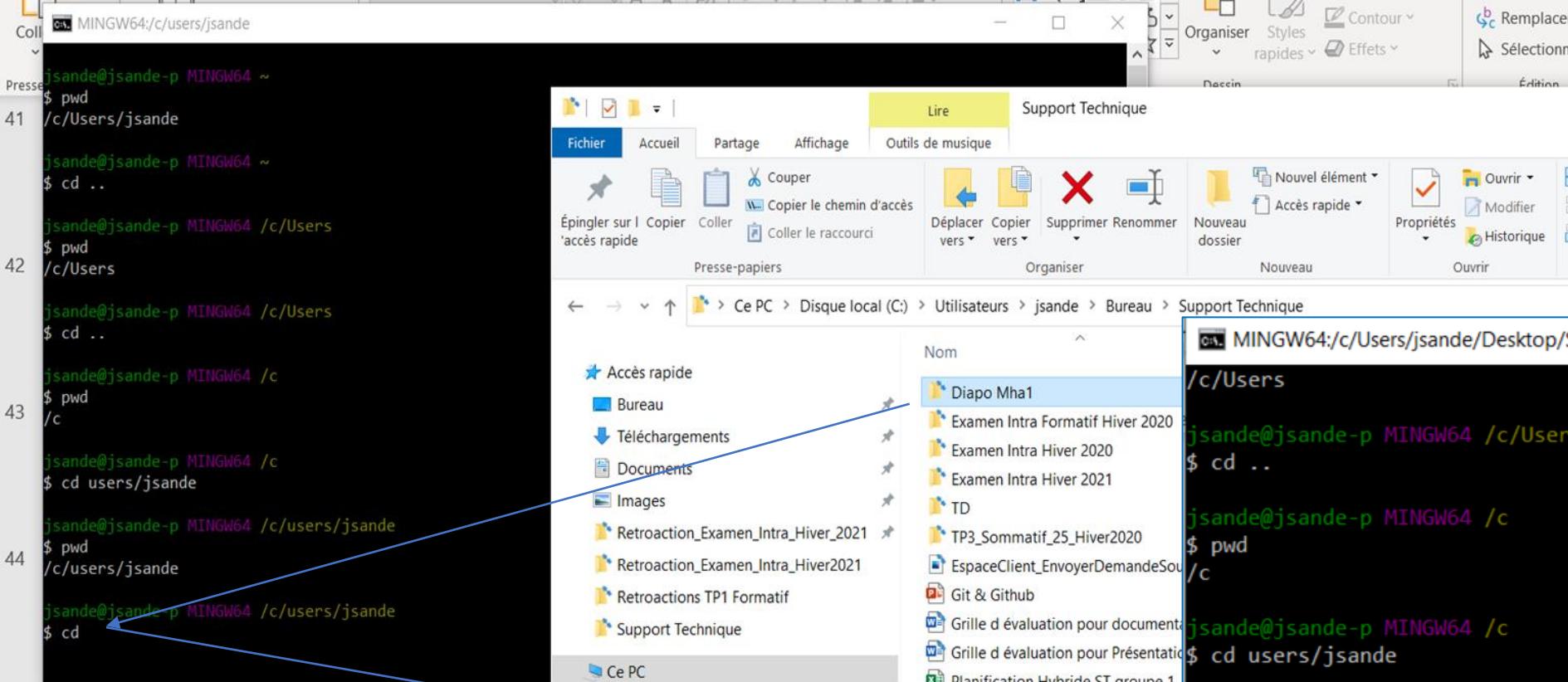
jsande@jsande-p MINGW64 ~
$ cd ..
jsande@jsande-p MINGW64 /c/Users
$ pwd
/c/Users

jsande@jsande-p MINGW64 /c/Users
$ cd ..
jsande@jsande-p MINGW64 /c
$ pwd
/c

jsande@jsande-p MINGW64 /c
$ cd users/jsande

jsande@jsande-p MINGW64 /c/users/jsande
$ pwd
/c/users/jsande

jsande@jsande-p MINGW64 /c/users/jsande
$ -
```



Je peux faire un drag-and-drop du dossier du mariage que je vous ai partagé.

Voici son contenu

```
MINGW64:/c/Users/jsande/Desktop/Support Technique/Diapo Mha1
/c/Users
jsande@jsande-p MINGW64 /c/Users
$ cd ..
jsande@jsande-p MINGW64 /c
$ pwd
/c
jsande@jsande-p MINGW64 /c
$ cd users/jsande
jsande@jsande-p MINGW64 /c/users/jsande
$ pwd
/c/users/jsande
jsande@jsande-p MINGW64 /c/users/jsande
$ cd "C:\Users\jsande\Desktop\Support Technique\Diapo Mha1"
jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
$ pwd
/c/Users/jsande/Desktop/Support Technique/Diapo Mha1
jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
$ ls
'Abidjan Farot.mp3'          Diapos_1_Joel_Ruth.pptx  'Fete de Noel.pptx'
'Chanson de Noel 1.mp3'       Diapos_2_Francis.pptx   PourToutCeQueTuEs.mp3
jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
$
```

Voici quelques commandes utiles sur git

- `$ git --version` nous dira si Git est installé sur notre ordinateur.
- `$ mkdir nom_nouveau_dossier` nous permet de créer un dossier
- `$ rmdir nom_nouveau_dossier` nous permet de supprimer le dossier
- `$ ls -la` permet d'afficher tous les fichiers et dossiers y compris ceux qui sont cachés.
- `$ git status` nous donne l'état actuel de notre environnement de travail. **Nous l'utiliserons beaucoup celle-là.**
- `$ clear` permet d'effacer tout ce que j'ai sur mon terminal.

MINGW64:/c/Users/jsande/Desktop/Support Technique/Diapo Mha1

```
jsande@jsande-p MINGW64 /c/users/jsande
$ pwd
/c/users/jsande

jsande@jsande-p MINGW64 /c/users/jsande
$ cd "C:\Users\jsande\Desktop\Support Technique\Diapo Mha1"

jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
$ pwd
/c/Users/jsande/Desktop/Support Technique/Diapo Mha1

jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
$ ls
'Abidjan Farot.mp3'      Diapos_1_Joel_Ruth.pptx  'Fete de Noel.pptx'
'Chanson de Noel 1.mp3'   Diapos_2_Francis.pptx    PourToutCeQueTuEs.mp3

jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
$ ls -all
total 30856
drwxr-xr-x 1 jsande 1049089      0 avr.  1 16:19  /
drwxr-xr-x 1 jsande 1049089      0 avr. 17 18:20  ../
-rw-r--r-- 1 jsande 1049089 3206789 juil. 21 2014 'Abidjan Farot.mp3'
-rw-r--r-- 1 jsande 1049089 2187736 avr.  1 00:43 'Chanson de Noel 1.mp3'
-rw-r--r-- 1 jsande 1049089 6773239 mars  31 23:35 Diapos_1_Joel_Ruth.pptx
-rw-r--r-- 1 jsande 1049089 7376102 juil. 25 2014 Diapos_2_Francis.pptx
-rw-r--r-- 1 jsande 1049089 6252445 avr.  1 16:19 'Fete de Noel.pptx'
-rw-r--r-- 1 jsande 1049089 6154240 mai   10  2008 PourToutCeQueTuEs.mp3
```

Avec l'ajout du mot-clé –all , j'ai un peu plus de détails sur les items affichés.

```
jsande@jsande-p MINGW64 ~/Desktop/Support Technique/Diapo Mha1
```

```
$
```

Créez vous un [Dossier Git](#) pas loin de votre dossier Javascript et rentrez la commande « [git](#) » pour être sûr de voir ce que je vois présentement.

Cela vous donne d'autres idée de commandes intéressantes et que nous utiliserons au fur et à mesure.

```
MINGW64:/c/Users/jsande/Documents/Dossiers_Git
$ pwd
/c/Users/jsande/Documents/Dossiers_Git

jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git
$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           [--super-prefix=<path>] [--config-env=<name>=<envvar>]
           <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone          Clone a repository into a new directory
  init           Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add            Add file contents to the index
  mv             Move or rename a file, a directory, or a symlink
  restore        Restore working tree files
  rm             Remove files from the working tree and from the index
  sparse-checkout Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)
  bisect         Use binary search to find the commit that introduced a bug
  diff           Show changes between commits, commit and working tree, etc
  grep           Print lines matching a pattern
  log            Show commit logs
  show           Show various types of objects
  status         Show the working tree status

grow, mark and tweak your common history
  branch        List, create, or delete branches
```

Initialisation de Git

- Après avoir installé Git sur son ordinateur, on peut très bien travailler en local. Mais avant tout, il faut l'initialiser/le configurer :

```
$ git config --global user.name alibaba  
$ git config --global user.email xxx@xxx.xx
```

- Grace à l'option global, cette command sera faite une seule fois pour toute.

```
$ git config --global -list pour confirmer que la configuration a bien été effectuée.  
$ git sans aucun paramètres, c'est une commande très-très utile.
```

- C'est une bonne idée d'activer les couleurs afin d'améliorer la lisibilité des différentes branches. A cet effet, lançons ces 3 lignes de commande :

- \$ git config --global color.diff auto
- \$ git config --global color.status auto
- \$ git config --global color.branch auto

- Par défaut, Git utilisera Vim comme éditeur et Vimdiff comme outil de merge. Vous pouvez les modifier en utilisant :

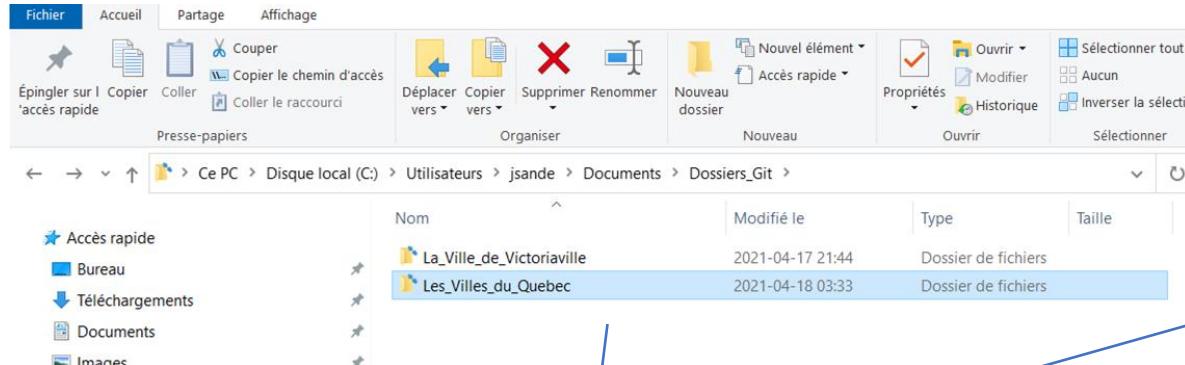
- \$ git config --global core.editor notepad++
- \$ git config --global merge.tool vimdiff

Comment créer un dépôt local ?

Comment créer un dépôt local ?

- Dans un logiciel de gestion de versions comme Git, une copie de projet est appelé : **un dépôt**.
- Il y a 2 façon de créer un dépôt local:
 1. En **créant un dépôt local vide** pour accueillir un nouveau projet ;
 2. En **clonant un dépôt distant**, c'est-à-dire rapatrier tout l'historique d'un dépôt distant en local, afin de pouvoir travailler par dessus.

1 Créer un dépôt local vide



```
jsande@jsande-p MINGW64 ~/documents/dossiers_git/La_Ville_de_Victoriaville (main)
$ pwd
/c/Users/jsande/documents/dossiers_git/La_Ville_de_Victoriaville

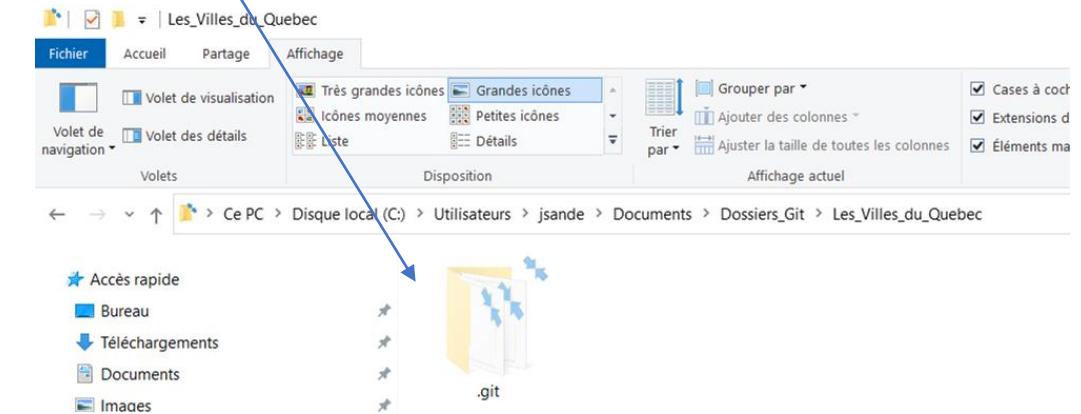
jsande@jsande-p MINGW64 ~/documents/dossiers_git/La_Ville_de_Victoriaville (main)
$ cd ..

jsande@jsande-p MINGW64 ~/documents/dossiers_git
$ cd Les_Villes_du_Quebec

jsande@jsande-p MINGW64 ~/documents/dossiers_git/Les_Villes_du_Quebec
```

1. Je vais créer un dossier que je vais appeler [Les_Villes_du_Quebec](#).
2. Je vais positionner Git Bash sur ce répertoire.
3. Je vais lancer la commande suivante:
`$ git init`

Un dossier caché `.git` sera créé



1 Créer un dépôt local vide

The screenshot shows a Windows File Explorer window with the following details:

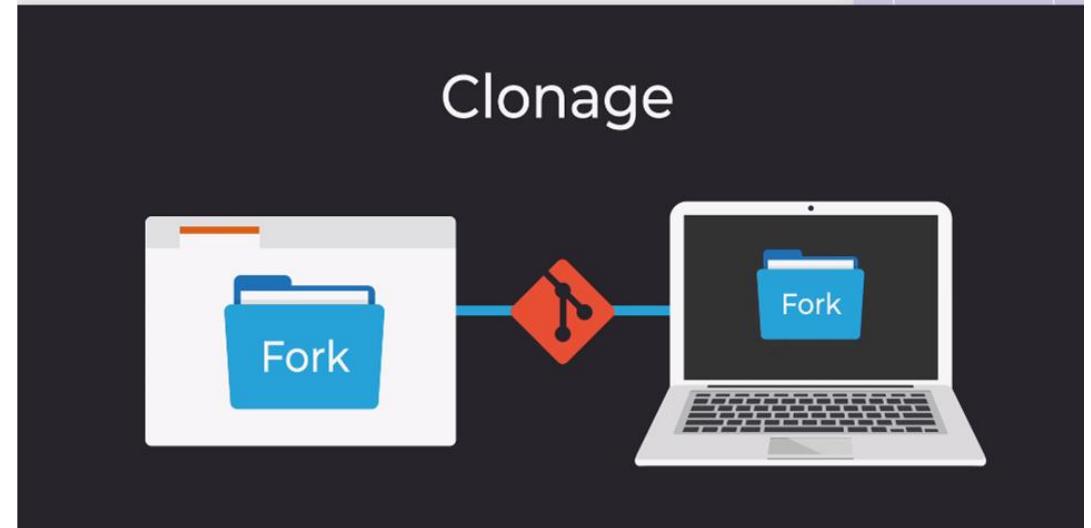
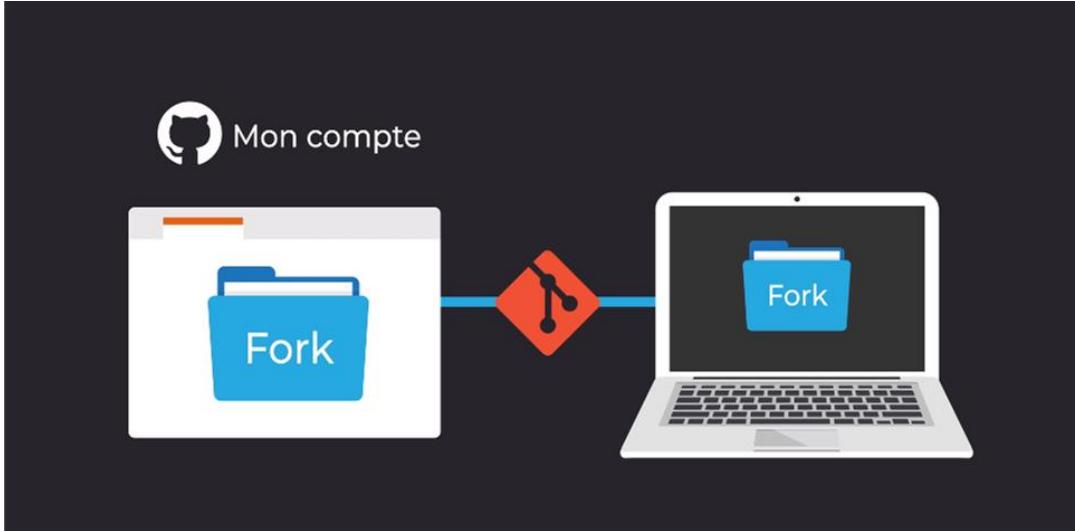
- Address Bar:** Les_Villes_du_Quebec
- Menu Bar:** Fichier, Accueil, Partage, Affichage
- Toolbars:** Volet de visualisation, Volet des détails
- View Options:** Très grandes icônes, Grandes icônes (selected), Icônes moyennes, Petites icônes, Liste, Détails
- Sort Options:** Grouper par, Trier par, Ajouter des colonnes, Ajuster la taille de toutes les colonnes
- Checkboxes:** Cases à cocher, Extensions d..., Éléments ma...
- Breadcrumbs:** Ce PC > Disque local (C:) > Utilisateurs > jsande > Documents > Dossiers_Git > Les_Villes_du_Quebec
- Left Sidebar:** Accès rapide (Bureau, Téléchargements, Documents, Images)
- Right Sidebar:** A yellow folder icon with blue arrows and a ".git" extension.

Text Box Content:

À partir de ce répertoire ([officiellement un dépôt Git à présent](#)) je peux glisser, créer toute sorte de dossier ou fichier ici : c'est ma branche principale.

Attention : je ne touche **PAS DU TOUT** au dossier [.git](#)

2 Cloner un dépôt distant depuis mon compte GitHub



Je crée un dépôt (repository) sur Github que je **fork** en local.

L'action de forker en local consiste à cloner ce dépôt.

On parle de **clonage** d'un dépôt.

Cloner le dépôt me permettra de télécharger mon projet et de le déposer en local pour pouvoir continuer à travailler dessus en local.

Cloner un dépôt distant depuis mon compte GitHub

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * Repository name *

 joel225 / Les_Villes_de_L_Ontario ✓

Great repository names are short and memorable. Need inspiration? How about [curly-happiness?](#)

Description (optional)

Dans ce projet, nous allons décrire les différentes villes de l'Ontario.

 Public
Anyone on the internet can see this repository. You choose who can commit.

 Private
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

Add a README file
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set  main as the default branch. Change the default name in your [settings](#).

[Create repository](#)

main

1 branch

0 tags

Go to file

Add file

Code

joel225 Initial commit

README.md

Initial commit

README.md

Les_Villes_de_L_Ontario

Dans ce projet, nous allons décrire les différentes villes de l'Ontario.

Clone

HTTPS SSH GitHub CLI

https://github.com/joel225/Les_Villes_de_L_Ontario

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

About

Dans ce projet, nous allons décrire les différentes villes de l'Ontario.

Readme

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

```
jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git
$ git clone https://github.com/joel225/Les_Villes_de_L_Ontario.git
Cloning into 'Les_Villes_de_L_Ontario'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

2 Cr ons-en un autre d p t distant sur GitHub

Nous allons cr er le d p t nomm  [La_Ville_de_Victoriaville](#) que nous allons par la suite importer en local.

The screenshot shows a GitHub repository page for the user 'joel225' with the repository name 'La_Ville_de_Victoriaville'. The repository is marked as a template. The main page displays the 'Code' tab, with the 'main' branch selected. A single file, 'README.md', is shown. The file was updated by 'joel225' 2 minutes ago, with a commit message 'Update README.md'. It has 1 contributor. The file content is as follows:

```
Ville_de_Victoriaville

Je vais dans ce dossier, vous parler de cette belle 茅cole situ e en plein centre de a ville de Victoriaville.
```

The page also includes standard GitHub navigation links like 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. At the bottom, there are links for 'Contact GitHub', 'Pricing', 'API', 'Training', 'Blog', and 'About'.

joel225 / La_Ville_de_Victoriaville Template

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code Use this template

joel225 Update README.md

README.md Update RE

README.md

Ville_de_Victoriaville

Je vais dans ce dossier, vous parler de cette belle école située en plein centre de a ville de Victoriaville.

Clone HTTPS SSH GitHub CLI
://github.com/joel225/La_Ville_de_Victoriaville Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop Download ZIP

9 minutes ago 2 commits

Readme

About

Je vais dans ce dossier, v cette belle ville située en de a ville de Victoriaville.

Readme

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Je vous recommande d'utiliser l'URL

Cliquez ici pour copier l'URL

Dans certaines versions c'est : clone or download

Puis coller la

```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git
$ git clone https://github.com/joel225/La_Ville_de_Victoriaville.git
```

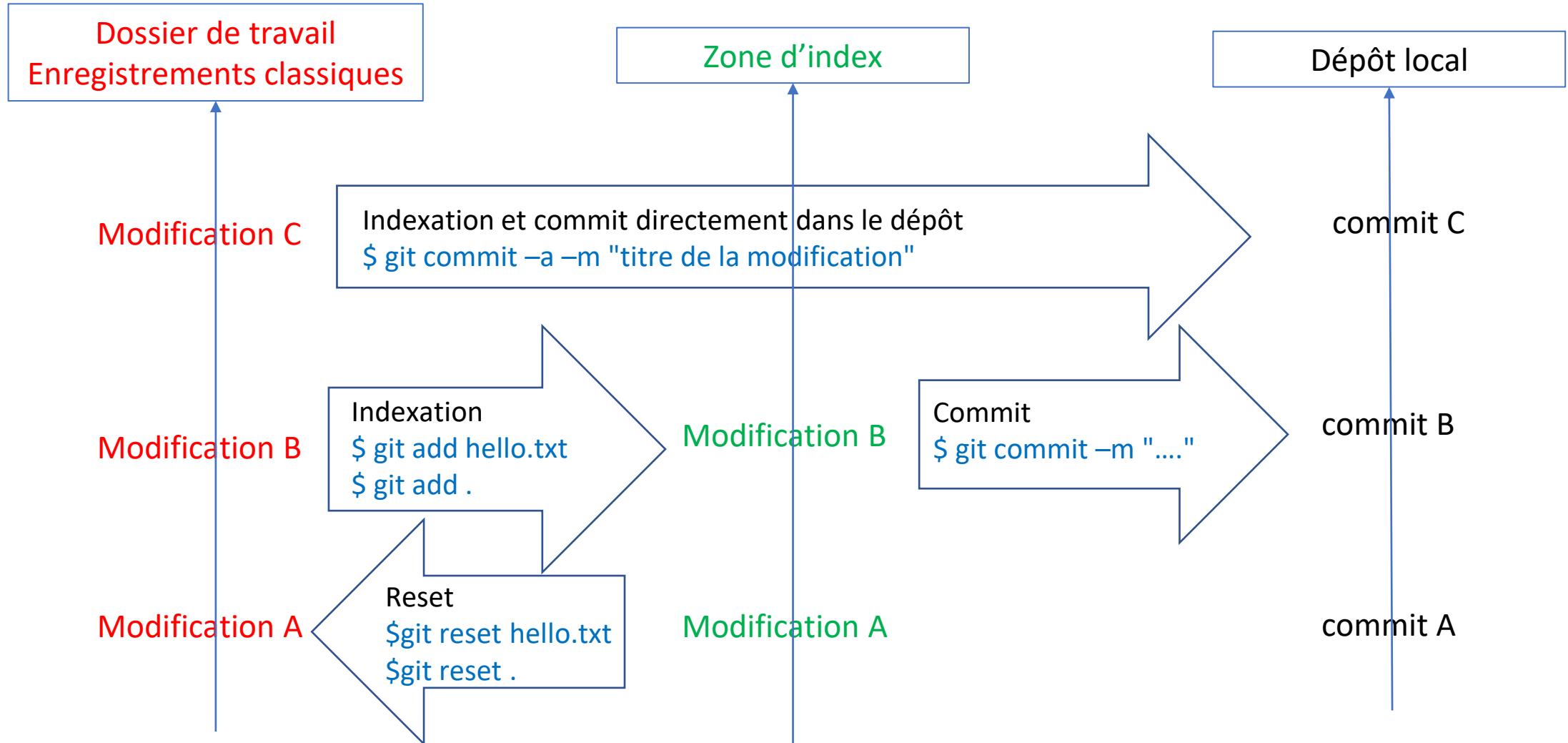
```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git
$ git clone https://github.com/joel225/La_Ville_de_Victoriaville.git
Cloning into 'La_Ville_de_Victoriaville'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
Resolving deltas: 100% (1/1), done.
```

```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git
$ -
```

Le dépôt distant
est à présent cloné
en local.

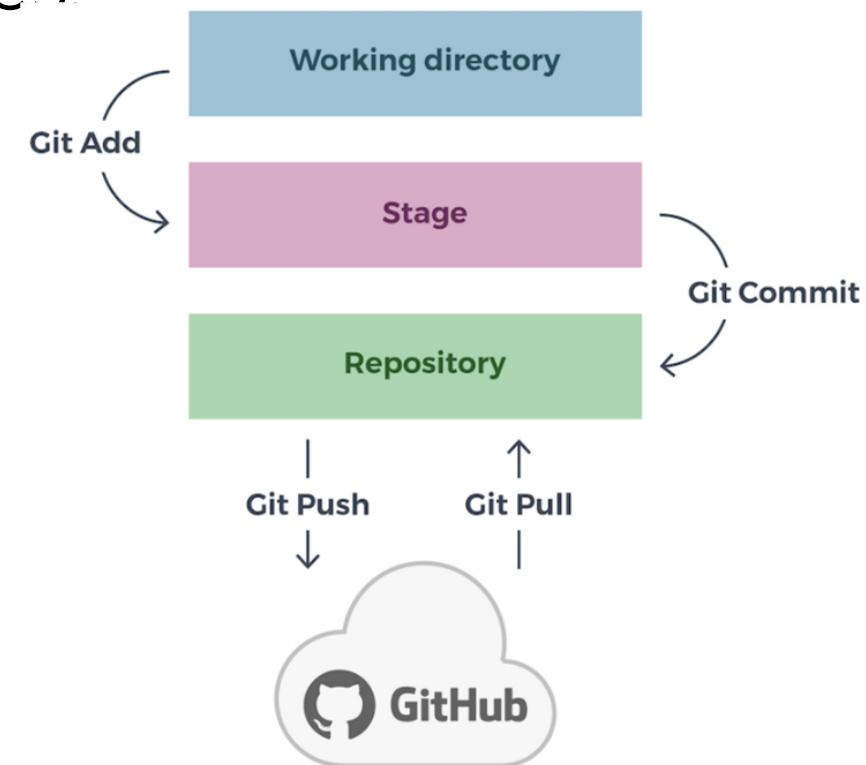
Ce PC > Disque local (C:) > Utilisateurs > jsande > Documents > Dossiers_Git > La_Ville_de_Victoriaville				
Nom	Modifié le	Type	Taille	
README	2021-04-17 21:44	Fichier source Mar...	1 Ko	

L'enregistrement des fichiers dans notre dépôt Git se déroule en 3 phases:



- En résumé, Git gère les versions de vos travaux locaux à travers 3 zones locales majeures

1. **Le répertoire de travail** (working directory/WD)
2. **L'index, ou stage** (nous préférerons le second terme)
3. **Le dépôt local** (Git directory/repository)



La commande `git status` vous permet de voir l'état de vos sauvegardes. Celle-ci vous met en garde que le fichier `README.md` a été modifié, mais n'a pas été commit (commit qui permettra de garder l'historique des changements.)

```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git/La_Ville_de_Victoriaville (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
            modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git/La_Ville_de_Victoriaville (main)
$
```

Pour faire le commit en local, il faut lancer cette commande. (`m = message`)

```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git/La_Ville_de_Victoriaville (main)
$ git commit -a -m "Nous avons ajouté du contenu à la ligne 4"
```

La moralité de cette histoire est qu'en ligne sur Github, « `commit` » sauvegarde et commit en même temps, tandis qu'en local il faut procéder à ces 2 étapes.

Je vais écrire
une 2e ligne que
je vais
« sauver » et
« commit »



```
C:\Users\jsande\Documents\Dossiers_Git\La_Ville_de_Victoriaville\README.md - Notepad++  
Fichier Édition Recherche Affichage Encodage Langage Paramètres Outils Macro Exécution Modules d'extension Documents ?  
README.md x  
1 # Ville_de_Victoriaville  
2 Je vais dans ce dossier, vous parler de cette belle école située en plein centre de ...  
3  
4 À partir de mon répertoire local, dans le fichier README, j'écris cette ligne.  
5  
6 Je vais écrire une 2e ligne. Victoriaville est une très belle ville.
```

```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git/La_Ville_de_Victoriaville (main)  
$ git commit -a -m "Je viens d'écrire une 2e ligne à la ligne 6"
```

Une 2^e façon de faire la même chose c'est de le faire en ces 2 étapes :
\$ git add README.md ou alors \$ git add .
\$ git commit -m "Je viens d'écrire une 2^e ligne à la ligne 6"

- J'apporte une petite modification à mon fichier et avant même de add, je tape la commande status :

\$ git status

- Mon fichier apparaît rouge
- Ensuite, je tape la commande diff :

\$ git diff README.md

- Je vois les modifications apportées, exactement comme le bouton history sur Github.

- J'indexe à présent mon fichier et je recommence l'opération (voir page suivante).

```
MINGW64:/c/Users/jsande/documents/dossiers_git/la_ville_de_victoriaville
jsande@jsande-p MINGW64 ~/documents/dossiers_git/la_ville_de_victoriaville (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

jsande@jsande-p MINGW64 ~/documents/dossiers_git/la_ville_de_victoriaville (main)
$ git diff README.md
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
diff --git a/README.md b/README.md
index 1efb2a0..2ef4f2a 100644
--- a/README.md
+++ b/README.md
@@ -9,4 +9,4 @@ Je viens de m'ajouter une 3e ligne pour vous faire voir un historique.

J'ajoute une 4e ligne au fichier README.md
On ajoute encore une ligne README.md
-Nous sommes rendu en saison d'été presque
+Nous sommes rendus en saison d'été presque
```

- J'index le fichier (les fichiers)
- Je vérifie le status.
- Je vois que **README.md** est indexé.
- Je lance à nouveau la commande diff:
- **\$ git diff**
- Je n'obtient rien car ce fichier a changé de position; il est dans la zone indexée. Je le retrouve en ajoutant **--cached** à la commande :
- **\$ git diff --cached**
- Dès que je le commit, il se déplacera à nouveau vers le dépôt cette fois-ci.

```

jsande@jsande-p MINGW64 ~/documents/dossiers_git/la_ville_de_victoriaville (main)
$ git add .
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory

jsande@jsande-p MINGW64 ~/documents/dossiers_git/la_ville_de_victoriaville (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   README.md

jsande@jsande-p MINGW64 ~/documents/dossiers_git/la_ville_de_victoriaville (main)
$ git diff

jsande@jsande-p MINGW64 ~/documents/dossiers_git/la_ville_de_victoriaville (main)
$ git diff --cached
diff --git a/README.md b/README.md
index 1efb2a0..2ef4f2a 100644
--- a/README.md
+++ b/README.md
@@ -9,4 +9,4 @@ Je viens de m'ajouter une 3e ligne pour vous faire voir un historique.

J'ajoute une 4e ligne au fichier README.md
On ajoute encore une ligne README.md
-Nous sommes rendu en saison d'été presque
+Nous sommes rendus en saison d'été presque

```

Les commandes log et config

Les commandes log et config

La commande log nous permet de voir l'historique de nos commits.

`$ git log`

Si cela n'avait pas été fait, ou si vous voulez apportez des modifications au niveau du nom d'utilisateur ou de l'email,

`git config --global user.name "votre nom d'utilisateur"`
`git config --global user.email xxx@xxx.xxx`

Pour en voir le résultat :

`$ git config --list`

```
jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git/La_Ville_de_Victoriaville (main)
$ git log
commit d8ff4d78af4feb01773dd6375fee2646998cda04 (HEAD -> main)
Author: Joel Sandé <Sande.Joel@cegepvicto.ca>
Date:   Sat Apr 17 23:06:49 2021 -0400

    Je viens d'ajouter une 3e ligne

commit fa8033465a6573958fdc342707e1de67b6377604
Author: Joel Sandé <Sande.Joel@cegepvicto.ca>
Date:   Sat Apr 17 22:48:24 2021 -0400

    Nous avons ajouté du contenu à la lfeigitclear alle 4

commit f53f01c631cd39e2f264b823b24d621517b6140e
Author: Joel Sandé <Sande.Joel@cegepvicto.ca>
Date:   Sat Apr 17 22:23:56 2021 -0400

    Nous avons ajouté du contenu à la ligne 4

commit b819e2111155f3c90e4840e01dccebeee727eae2 (origin/main, origin/HEAD)
Author: joel225 <33364732+joel225@users.noreply.github.com>
Date:   Sat Apr 17 21:22:51 2021 -0400

    Update README.md

commit 075aae3f60a04c08c4f2908c09a910509cf2abbf
Author: joel225 <33364732+joel225@users.noreply.github.com>
Date:   Sat Apr 17 21:15:04 2021 -0400

    Initial commit

jsande@jsande-p MINGW64 ~/Documents/Dossiers_Git/La_Ville_de_Victoriaville (main)
$
```

Transférer les changements de Git vers
Github

Transférer les changements de Git vers Github

À présent, je veux [envoyer](#) toutes ces mises-à-jours [en ligne](#). Je me souviens que mon fichier README.md ne contenait pas grand-chose. Voir aussi quelques diapos plus-haut au besoin.

The screenshot shows a GitHub repository interface. At the top, there are navigation buttons: 'main' (with a dropdown arrow), '1 branch' (with a dropdown arrow), '0 tags' (with a dropdown arrow), 'Go to file', 'Add file', 'Code' (with a dropdown arrow), and a green 'Use this template' button. To the right of these is an 'About' section with a summary: 'Je vais dans ce dossier, vous parler de cette belle ville située en plein centre de a ville de Victoriaville.' Below this is a 'Readme' link. Further down are sections for 'Releases' (with a note 'No releases published' and a 'Create a new release' link) and 'Packages'. The main content area displays a commit history and a file viewer for 'README.md'. The commit history shows one commit from 'joel225' titled 'Update README.md' made 2 hours ago. The file viewer shows the content of 'README.md': 'Ville_de_Victoriaville' and the descriptive text 'Je vais dans ce dossier, vous parler de cette belle école située en plein centre de a ville de Victoriaville.'

Voici d'ailleurs le contenu actuel de mon fichier README.md en ligne.

La_Ville_de_Victoriaville / README.md in main

[Edit file](#) [Preview changes](#)

```
1 # Ville_de_Victoriaville
2 Je vais dans ce dossier, vous parler de cette belle école située en plein centre de a ville de Victoriaville.
3
```

La commande push

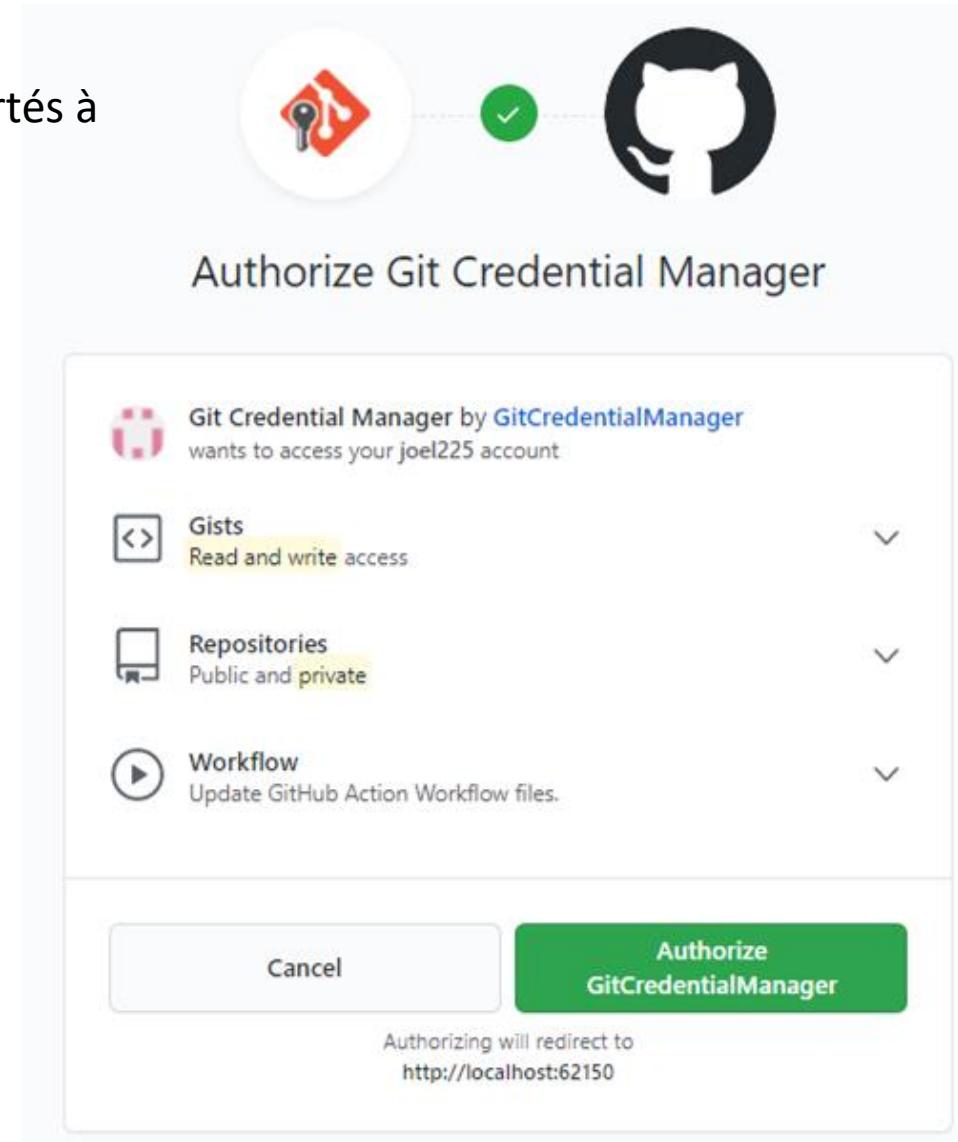
À partir de mon dépôt local, je **push** vers **origin**, tous les changements apportés à ma **branche courante**, qui ici est la main.

```
jsande@jsande-p MINGW64 ~/documents/dossiers_git/La_Ville_de_Victoriaville (main)
$ git push origin main
info: please complete authentication in your browser...
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 1001 bytes | 500.00 KiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/joel225/La_Ville_de_Victoriaville.git
  b819e21..d8ff4d7  main -> main

jsande@jsande-p MINGW64 ~/documents/dossiers_git/La_Ville_de_Victoriaville (main)
$
```

Vous aurez peut-être cette phrase d'erreur : si c'est le cas, je vous mets ce lien qui vous donnera la réponse selon le cas.

<https://stackoverflow.com/questions/4181861/message-src-refspec-master-does-not-match-any-when-pushing-commits-in-git>



La_Ville_de_Victoriaville / README.md in [main](#)[Edit file](#)[Preview changes](#)

```
1 # Ville_de_Victoriaville
2 Je vais dans ce dossier, vous parler de cette belle école située en plein centre de a ville de Victoriaville.
3
4 À partir de mon répertoire local, dans le fichier README, j'écris cette ligne.
5
6 Je vais écrire une 2e ligne. Victoriaville est une très belle ville.
7
8 Je viens de m'ajouter une 3e ligne pour vous faire voir un historique.
```

Le fichier est ainsi mis-à-jour

Et en voici l'historique des versions tel que nous l'avions en local.

main ▾

Commits on Apr 17, 2021

Je viens d'ajouter une 3e ligne

Joel Sandé authored and Joel Sandé committed 2 hours ago

 [d8ff4d7](#)

Nous avons ajouté du contenu à la ligne 4

Joel Sandé authored and Joel Sandé committed 3 hours ago

 [f880334](#)

Nous avons ajouté du contenu à la ligne 4

Joel Sandé authored and Joel Sandé committed 3 hours ago

 [f53f01c](#)

Update README.md

joel225 committed 4 hours ago

Verified [b819e21](#)

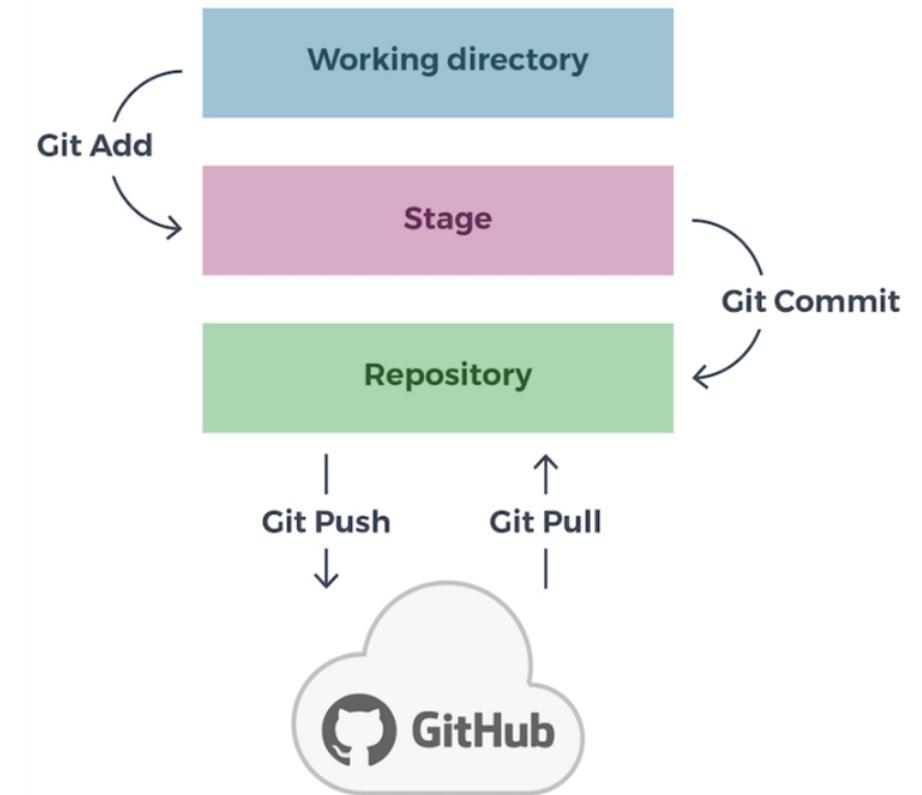
Initial commit

joel225 committed 4 hours ago

Verified [075aae3](#)

Question sur le Push

- Basé sur la commande précédente, quelle serait la commande pour **push** les mises-à-jour apportées en local à une branche 2 quelconque nommée **les_parcs** ?
-
- Remarque : tous les fichiers/dossiers de ma branche cible sont téléportés en même temps dans ce push sur GitHub .



Manipuler les branches en local

[joel225 / Les_Villes_de_L_Ontario](#) Private [Unwatch](#)

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)

[main](#) ▼ [2 branches](#) [0 tags](#) [Go to file](#) [Add file](#) [Code](#) ▼

 joel225 2e ligne du README Annonce d'un 2e projet pour le Quebec 57b0434 3 days ago [2 commits](#)

 README.md 2e ligne du README Annonce d'un 2e projet pour le Quebec 3 days ago

README.md [Edit](#)

Les_Villes_de_L_Ontario

Dans ce projet, nous allons décrire les différentes villes de l'Ontario. Et ensuite, dans un autre projet, nous procéderons pour le Québec

About [⚙️](#)
Dans ce projet, nous allons décrire les différentes villes de l'Ontario.

[Readme](#)

Releases
No releases published [Create a new release](#)

Packages
No packages published [Publish your first package](#)

Je vais me créer un dépôt que je vais appeler [Les_Villes_de_L_Ontario](#)

Dans le processus de description des villes, je vais me créer une 2^e branche que je vais appeler

Toronto

The screenshot shows a GitHub repository page. At the top, the repository name is 'joel225 / Les_Villes_de_L_Ontario' with a 'Private' badge. Below the name is a navigation bar with links: Code (highlighted), Issues, Pull requests, Actions, Projects, Security, Insights, and Settings.

In the main content area, there's a summary bar showing 'toronto' as the active branch, '2 branches', and '0 tags'. It also indicates that the branch is '5 commits ahead, 1 commit behind main'. Buttons for 'Go to file', 'Add file', and 'Code' are present. To the right, there's an 'About' section with a description: 'Dans ce projet, nous allons décrire les différentes villes de l'Ontario.' Below this is a 'Readme' section with a link. On the far right, sections for 'Releases' (no releases published) and 'Packages' (no packages published) are shown.

The 'Code' section displays a list of commits:

Author	Message	Date	Commits
joel225	une 3e ligne a été ajoutée au fichier Toronto.txt	2481946 3 days ago	6 commits
	README.md	J'ai créé le fichier Toronto.txt	3 days ago
	Toronto.txt	une 3e ligne a été ajoutée au fichier Toronto.txt	3 days ago

The 'README.md' file content is displayed below:

Les_Villes_de_L_Ontario

Dans ce projet, nous allons décrire les différentes villes de l'Ontario.

Nous allons donc commencer par les villes les plus Peuplées vers les villes les moins peuplées. Nous allons commencer par Toronto, qui est incontestablement la plus grande et plus riche ville du Canada. En quoi est-ce que Toronto est la plus grande ville du Canada ?

Manipuler les branches en local

- La commande `git branch` nous permet de voir toutes les branches présentes dans notre dépôt. Pour l'instant, il y a juste la branche `master`
- Dans un travail de groupe, il est souvent préférable de créer une branche pour une modification. La création prend 30 secondes et vous économise beaucoup de temps de galère si vous faites des bêtises sur votre branche master.
- Nous avons donc notre branche `master`, et nous souhaitons maintenant commencer par décrire la ville de Toronto. Pour cela, on crée la branche `Toronto` :
`git branch Toronto`
- Lançons à nouveau la commande `git branch`; On devrait voir 2 branches

Manipuler les branches en local

- Pour travailler sur la branche Toronto : `git checkout Toronto`
- Lançons à nouveau la commande `git branch`

Nous allons à présent commit le changement apporté à la branche Toronto avec la description de l'enregistrement pour pouvoir le retrouver.

Pour supprimer la branche Toronto :
`Git branch -d Toronto`

```
jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git
$ cd Les_Villes_de_L_Ontario

jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git/Les_Villes_de_L_Ontario (main)
$ git branch Toronto

jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git/Les_Villes_de_L_Ontario (main)
$ git branch
  Toronto
* main

jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git/Les_Villes_de_L_Ontario (main)
$ git checkout Toronto
Switched to branch 'Toronto'

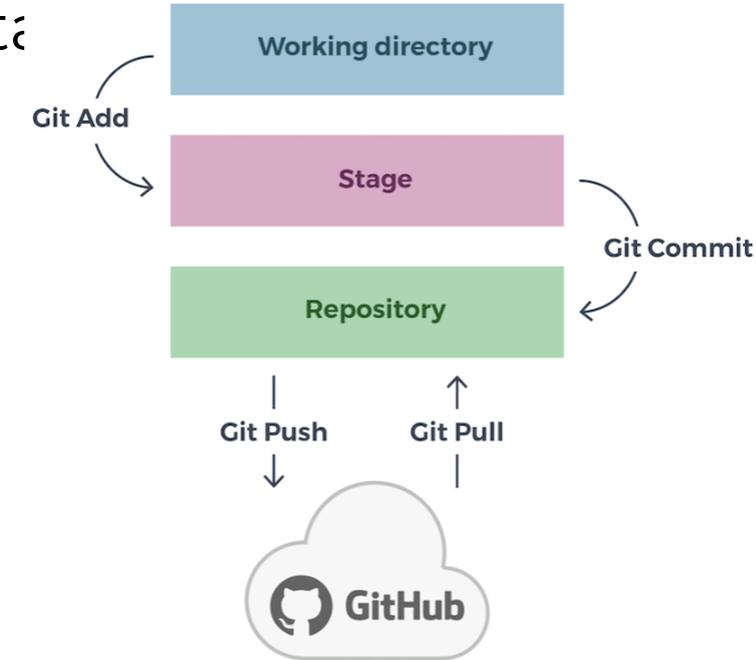
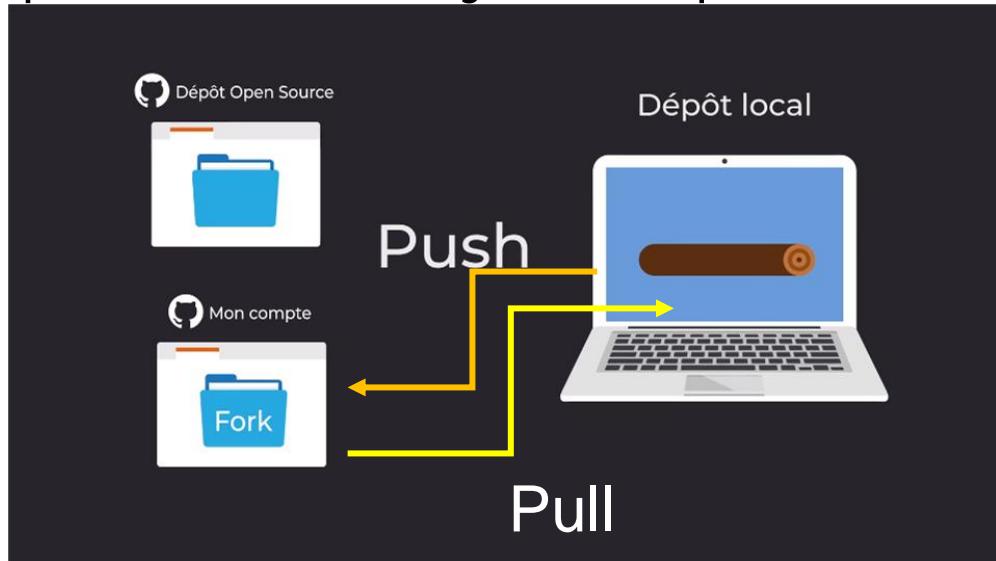
jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git/Les_Villes_de_L_Ontario (Toronto)
$ git branch
* Toronto
  main

jsande@jsande-p MINGW64 /c/users/jsande/documents/dossiers_git/Les_Villes_de_L_Ontario (Toronto)
$ git commit -m "introduction partant des grandes villes vers les petites"
```

Synchroniser :
Différence entre `git push` et `git pull`

Synchroniser : Différence entre `git push` et `git pull`

- La commande **Git push** permet d'envoyer les modifications que l'on a réalisées en local sur une branche vers le dépôt distant(github) sur la branche correspondante, alors que la commande **Git pull** permet de faire l'inverse avec le même principe (c'est la branche sur laquelle je me positionne en local qui est mise-à-jour depuis le dépôt dist)



Projet P5js sur Github

- Vous faites du développement Web en ce moment. Vous avez donc probablement beaucoup de fichiers sur lesquels vous travaillez en même temps. Voici donc pour vous une illustration de projet P5js fait et sauvegardé sur Github.



Différence entre un Merge et un Rebase

