
TP N°1 : Git et GitHub

SOMMAIRE

I- Objectifs :	3
II- Outils utilisés :	3
II- Gestion de versionning avec Git	3
a. Installation de Git au niveau d'eclipse	3
b. Configuration de Git	4
c. Création et utilisation du Repository	6
1. Création d'un nouveau repository	6
2. Création du fichier .gitignore	7
3. Ajouter votre projet dans Git Repository	8
4. Ouvrir la perspective Git	9
5. Ouvrir la vue Git Staging	10
6. Ouvrir la vue Git History	11
7. Supprimer un fichier du Repository	12
d. Cloner un repository existant	13
e. Utiliser un Repository Remote (exemple GITHUB)	21
1. Création d'un compte GITHUB et créer un nouveau Repository	21
2. Cloner le Repository au niveau d'Eclipse	25

I- Objectifs :

- ✓ Utiliser Git moyennant Eclipse.
- ✓ Créer un Repository.
- ✓ Cloner un Repository existant.
- ✓ Utiliser Github.
- ✓ Cloner un Repository sur GitHub.
- ✓ Commit et Push.
- ✓ Visualiser l'historique des commit.

II- Outils utilisés :

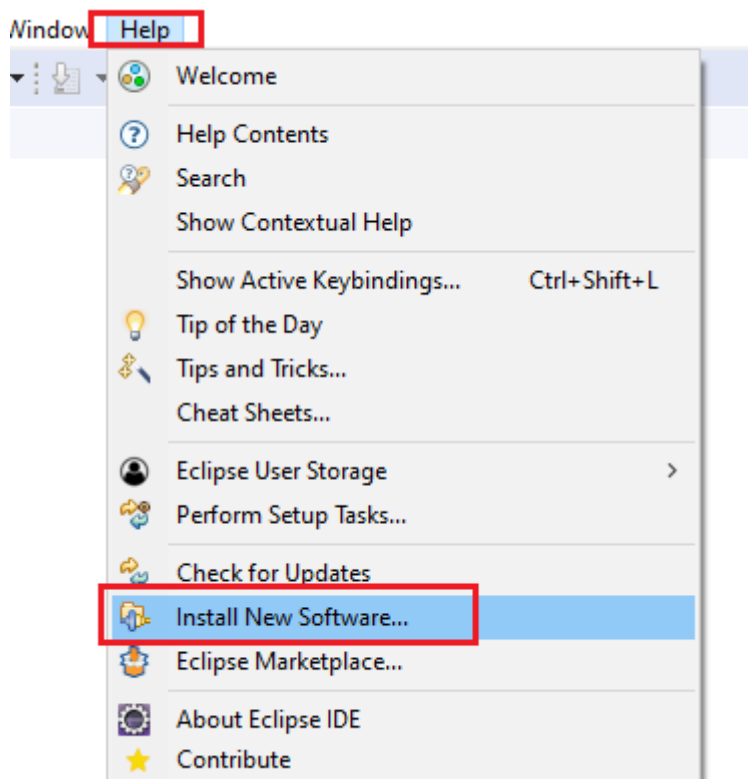
- ✓ Eclipse avec le plugin Maven
- ✓ Connection Internet pour télécharger le plugin de Git.
- ✓ Compte GitHub.

II- Gestion de Versionning avec Git

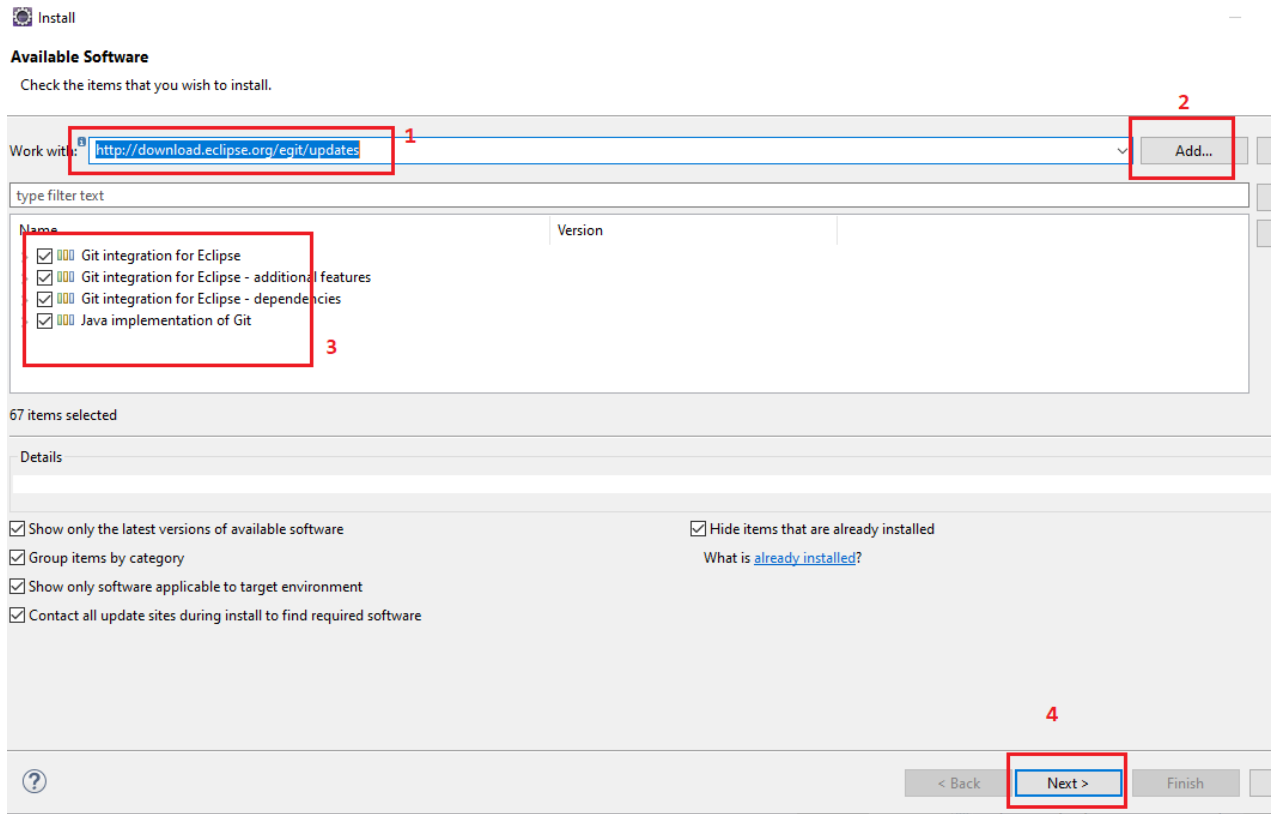
a. Installation de Git au niveau d'eclipse

Les nouvelles versions d'eclipse contiennent le plugin de Git. Au cas où votre Eclipse ne contient pas Git, il suffit de l'installer en suivant les étapes suivantes :

- Cliquer sur Help -> Install New Software ... :



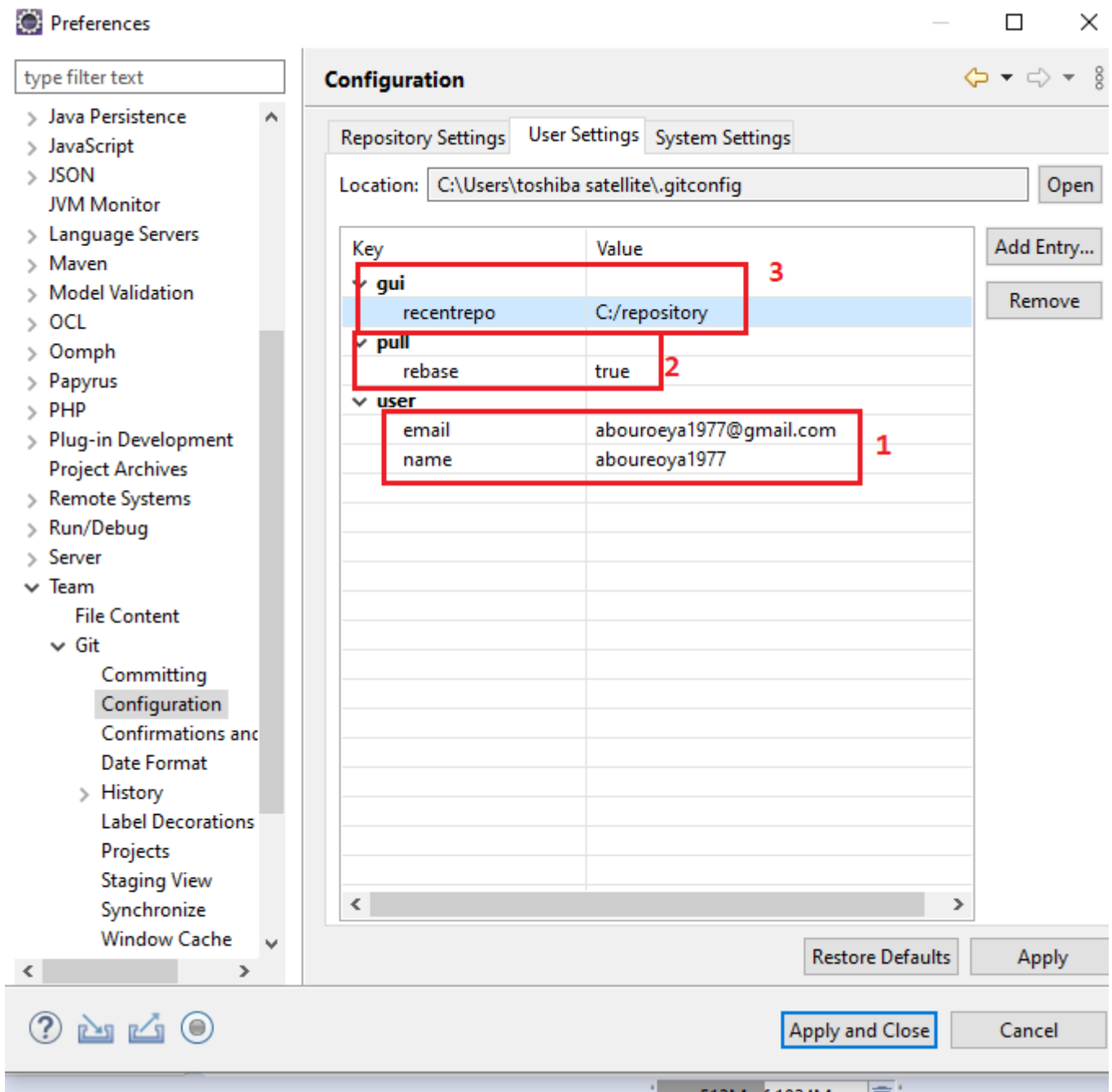
La fenêtre suivante s'affiche :



1. Entrer le site : <http://download.eclipse.org/egit/updates>
2. Cliquer sur Add...
3. Cocher les fonctionnalités de Git à installer
4. Cliquer sur Next>.

b. Configuration de Git

- Cliquer sur **Window -> Preferences -> Team -> Git -> Configuration** et suivre les étapes suivantes :

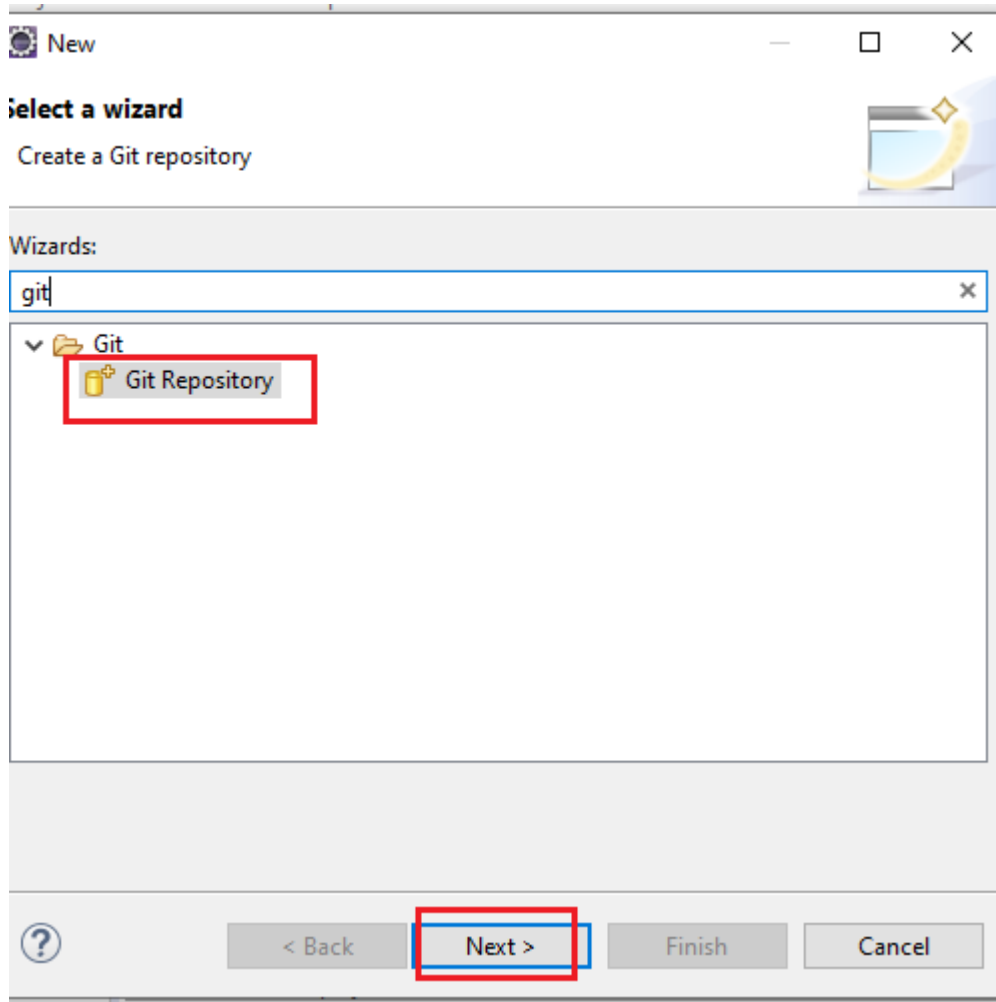


1. Entrer l'adresse mail (user.email) et le nom (user.name) ;
2. Entrer true dans rebase.
3. Entrer le dossier de votre Repository.

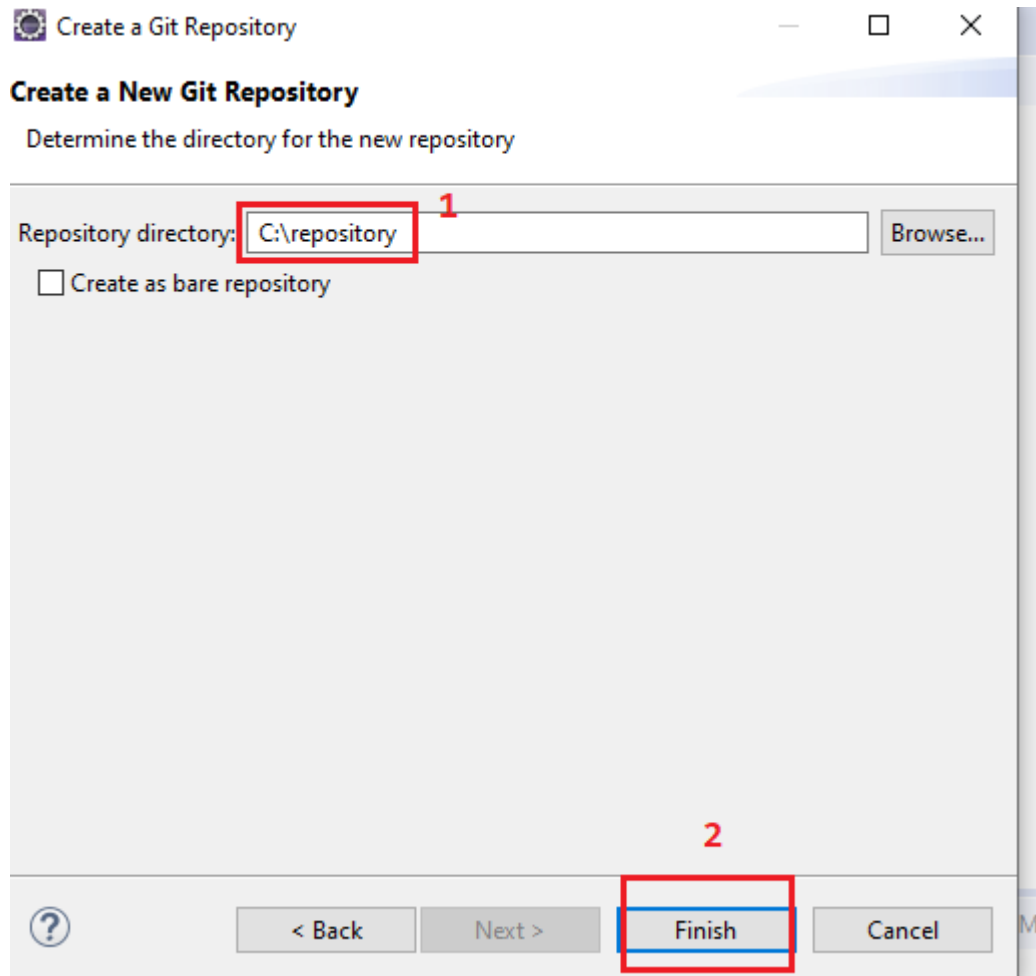
c. Création et utilisation du Repository

1. Création d'un nouveau repository

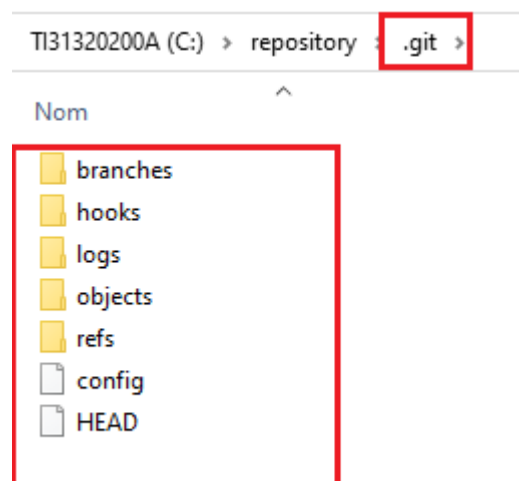
- Cliquer sur **File ->new->Git->Git Repository** :



- Cliquer sur **Next >**, la fenêtre suivante s'affiche :

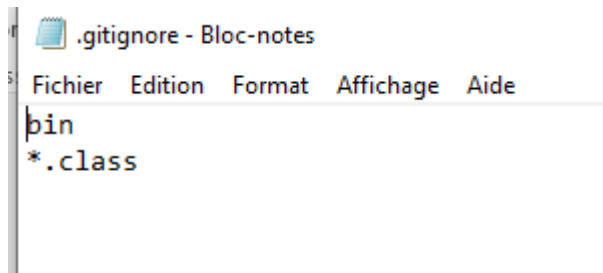


1. Entrer le chemin de votre repository.
2. Cliquer sur Finish. Le dossier caché .git est créé :



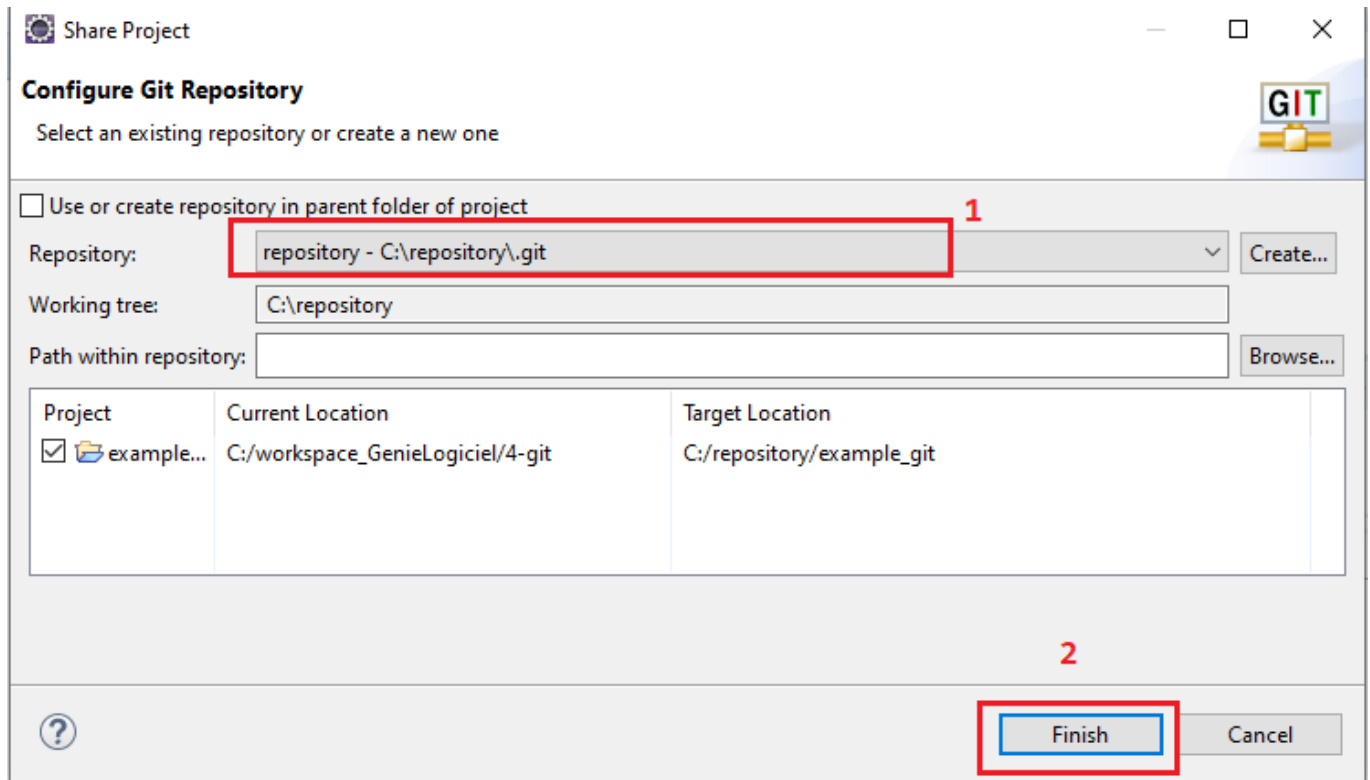
2. Création du fichier .gitignore

- Créer le fichier .gitignore dans le dossier racine de votre repository. Ensuite copier les deux lignes suivantes pour que Git ignore lors des commit le dossier bin et les fichiers .class :



3. Ajouter votre projet dans Git Repository

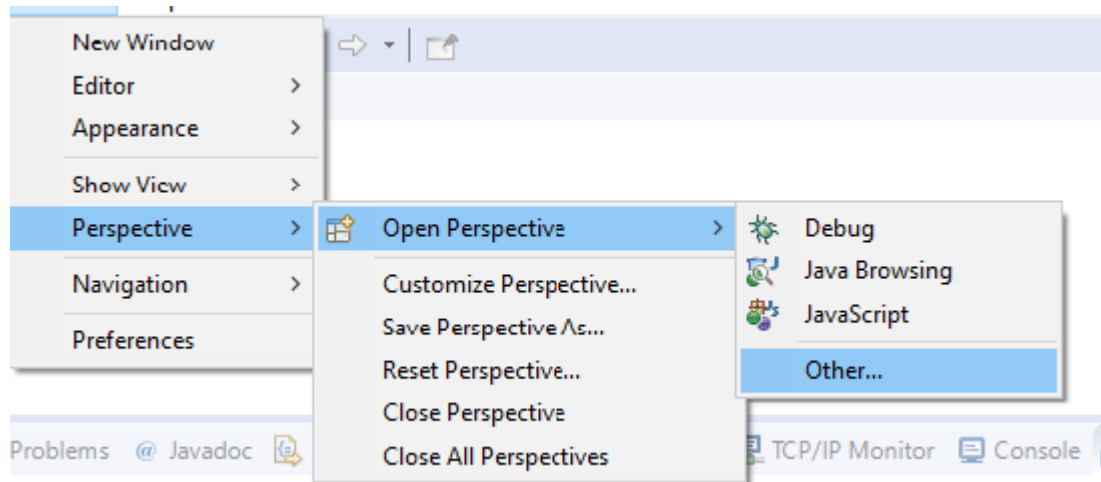
- Sélectionner le projet que vous voulez ajouter au Git Repository (par exemple : exemple_git).
- Cliquer à droite de la souris ensuite **Team -> Share Projet ...** :



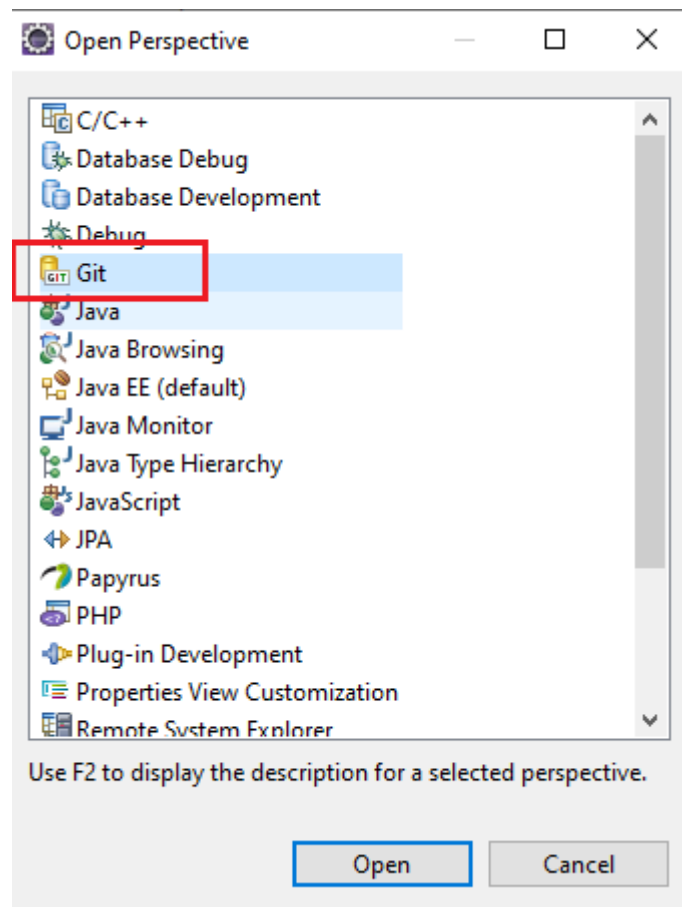
1. Entrer votre repository.
2. Cliquer sur Finish.

4. Ouvrir la perspective Git

- Cliquer sur **Window -> Perspective -> Open Perspective -> Other...** comme le montre l'écran suivant :



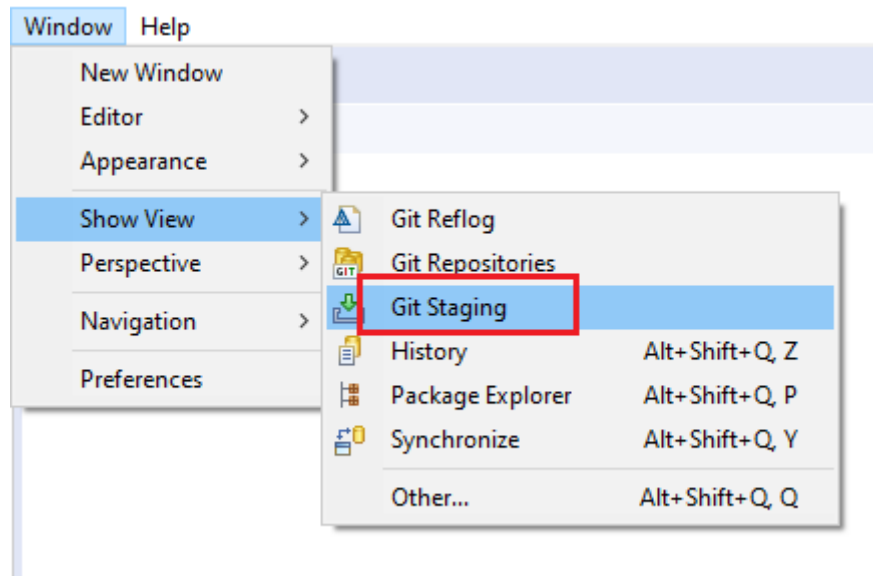
La fenêtre suivante s'affiche :



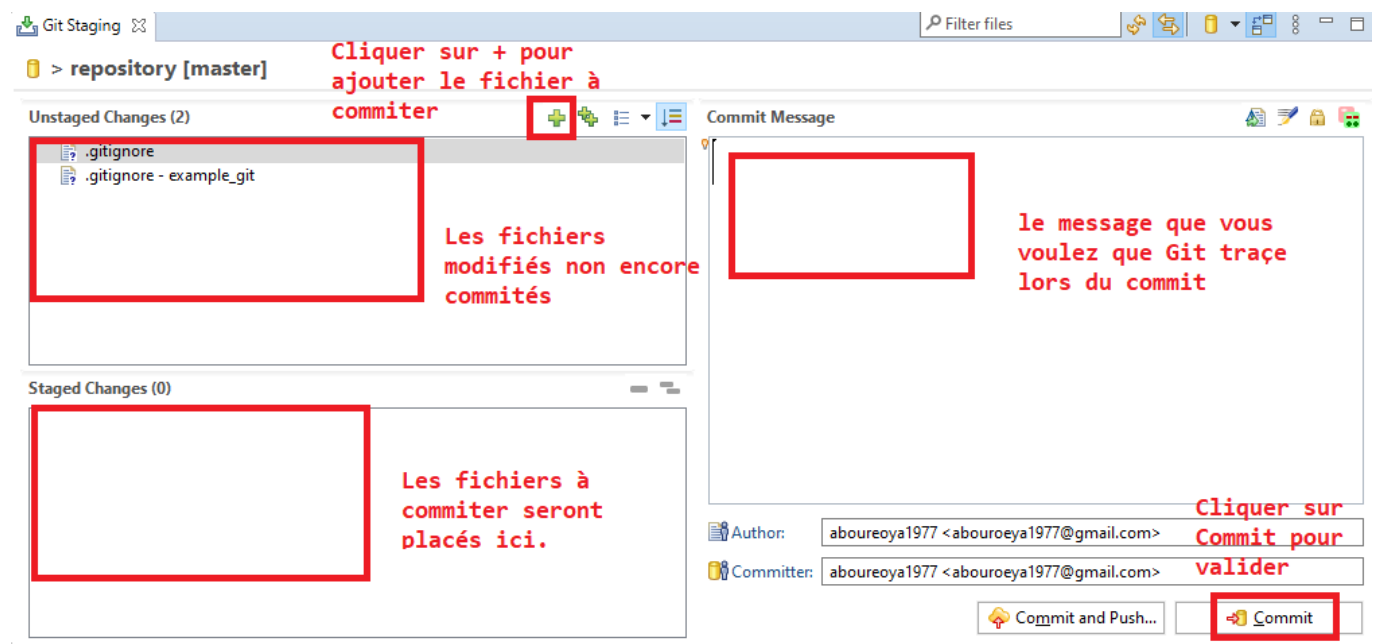
Sélectionner Git et cliquer sur Open.

5. Ouvrir la vue Git Staging

- Cliquer sur Window -> Show View -> Git Staging comme le montre l'écran suivant :

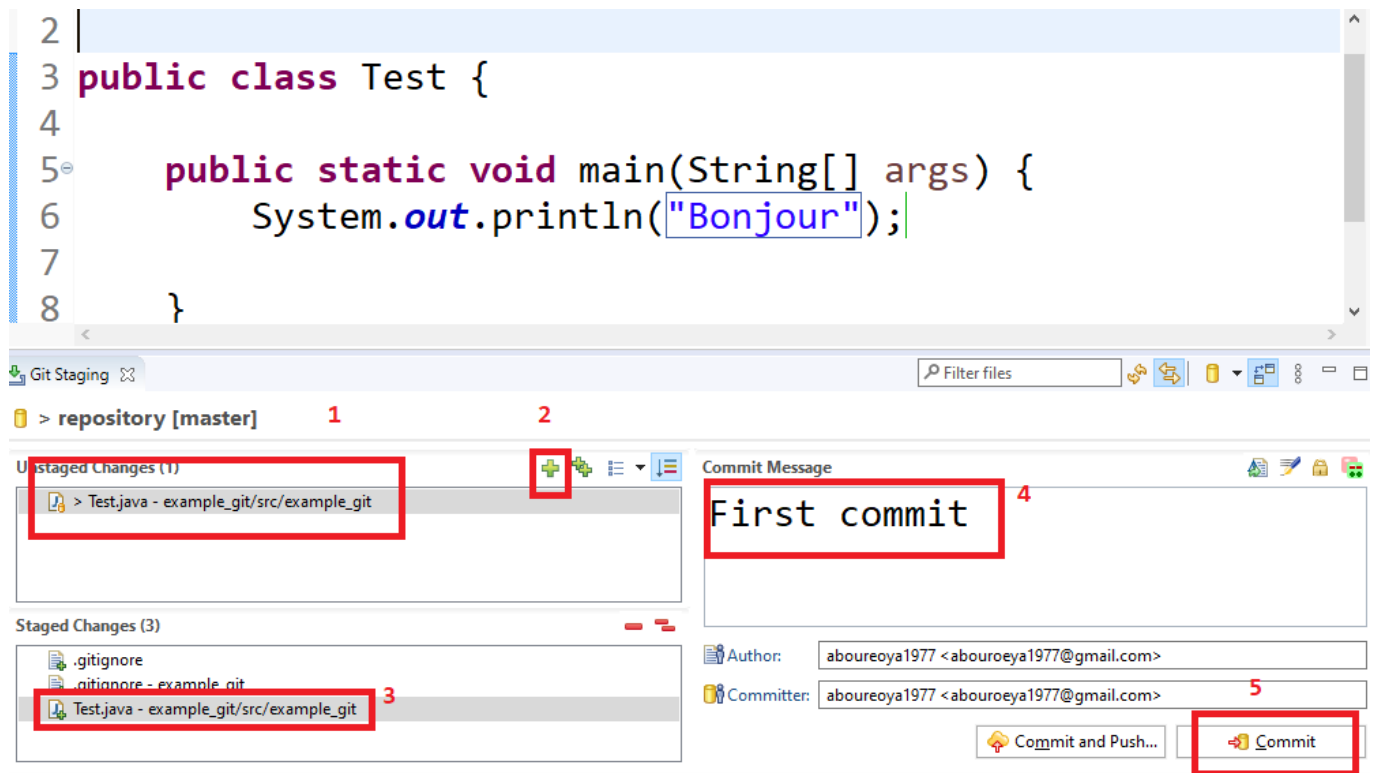


La fenêtre suivante s'affiche :



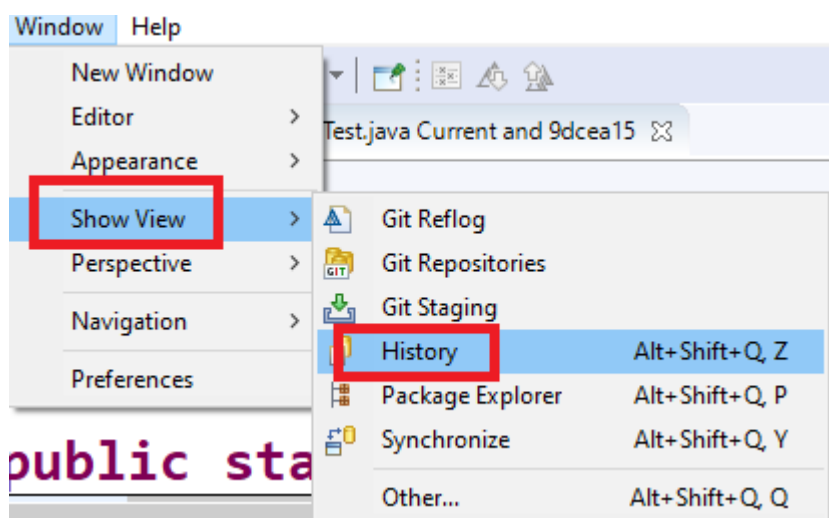
Voir les explications de la vue « Git Staging ».

- Créer une nouvelle classe Test et utiliser la vue « Git Staging » pour l'ajouter à Git en suivant les étapes suivantes :



1. Observer que la classe Test a été ajouté automatiquement.
2. Sélectionner la classe Test et cliquer sur + pour l'ajouter à Git.
3. Vérifier que la classe Test a été ajoutée dans « Staged Changes ».
4. Entrer votre message pour le commit.
5. Cliquer sur Commit pour valider.
6. **Ouvrir la vue Git History**

- Cliquer sur Window->Show View->History comme le montre l'écran suivant :



La fenêtre suivante s'affiche :

History Synchronize Git Staging Git Reflog Properties					
File: example_git/src/example_git/Test.java [repository]					
Id	Message	Author	Authored Date	Committer	Committed Date
86d96a2	master (HEAD) secont commit	aboureoya1977	2 minutes ago	aboureoya1977	2 minutes ago
9dcea15	First commit	aboureoya1977	2 minutes ago	aboureoya1977	2 minutes ago
1a93d4b	First commit	aboureoya1977	29 minutes ago	aboureoya1977	29 minutes ago
403e3e4	first commit	aboureoya1977	18 hours ago	aboureoya1977	18 hours ago

Pour visualiser les modifications de chaque commit, il suffit de cliquer sur la ligne qui concerne le commit en question. L'écran suivant s'affiche :

ocal: Test.java

```

3 public class Test {
4
5     public static void main(St
6     System.out.println("Bo
7
8     }
9

```

Test.java 9dcea15 (aboureoya1977)

```

3 public class Test {
4
5     public static void main(
6     // TODO Auto-generat
7
8     }
9

```

History Synchronize Git Staging Git Reflog Properties
File: example_git/src/example_git/Test.java [repository]

d	Message	Author	Authored Date	Committer	Committed Date
86d96a2	master (HEAD) secont commit	aboureoya1977	2 minutes ago	aboureoya1977	2 minutes ago
9dcea15	First commit	aboureoya1977	2 minutes ago	aboureoya1977	2 minutes ago
1a93d4b	First commit	aboureoya1977	29 minutes ago	aboureoya1977	29 minutes ago
403e3e4	first commit	aboureoya1977	18 hours ago	aboureoya1977	18 hours ago

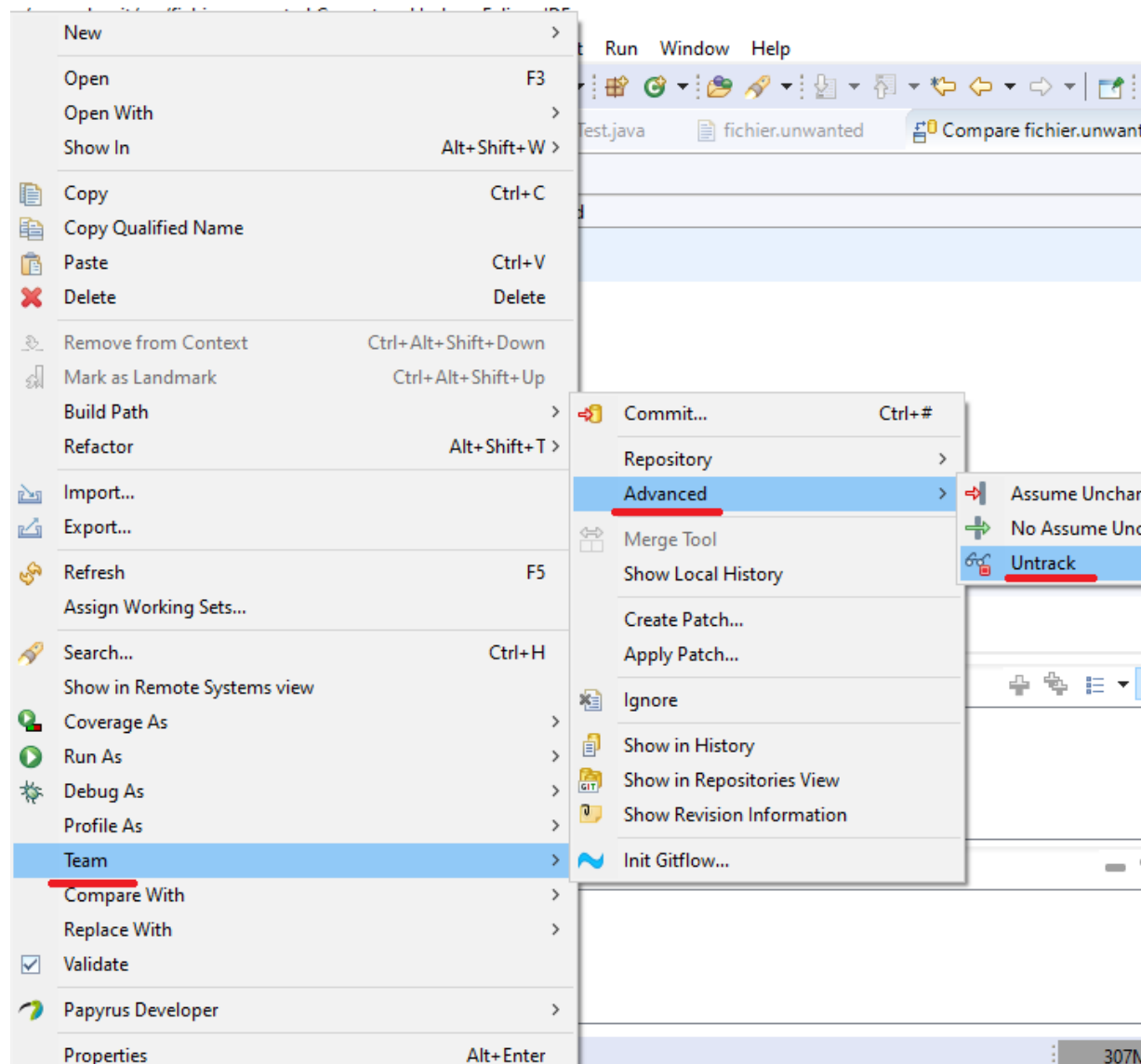
commit 9dcea15ea3f4d84fe89aa607848e9f30f8af9c
author: aboureoya1977 <aboureoya1977@gmail.com>

.gitignore
example_git/.gitignore
example_git/src/example_git/Test.java

7. Supprimer un fichier du Repository

Supposons que vous avez comité par erreur un fichier (par exemple : fichier.unwanted). Pour le supprimer du repository, suivre les étapes suivantes :

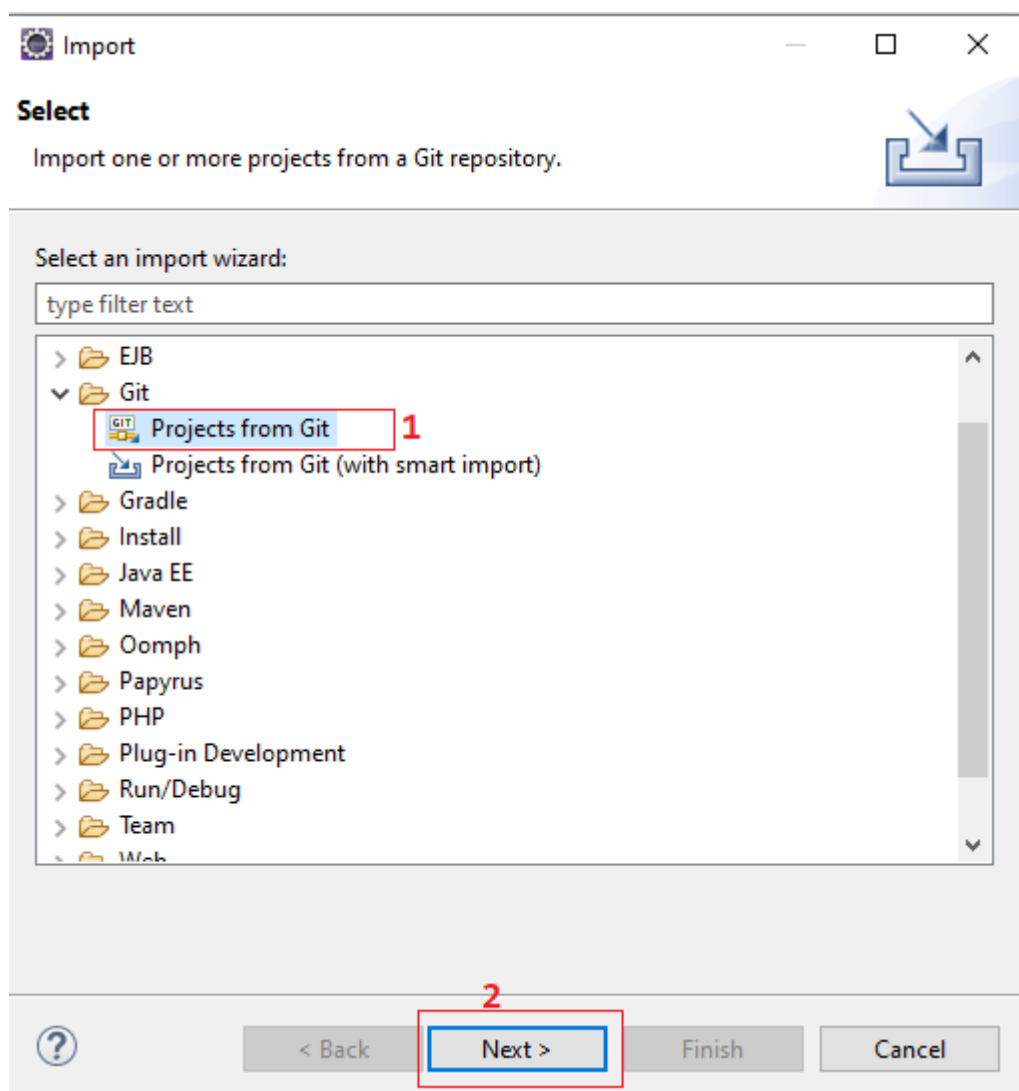
- Cliquer à droite de la souris, ensuite **Team->Advanced->Untrack** comme le montre l'imprime écran suivant :



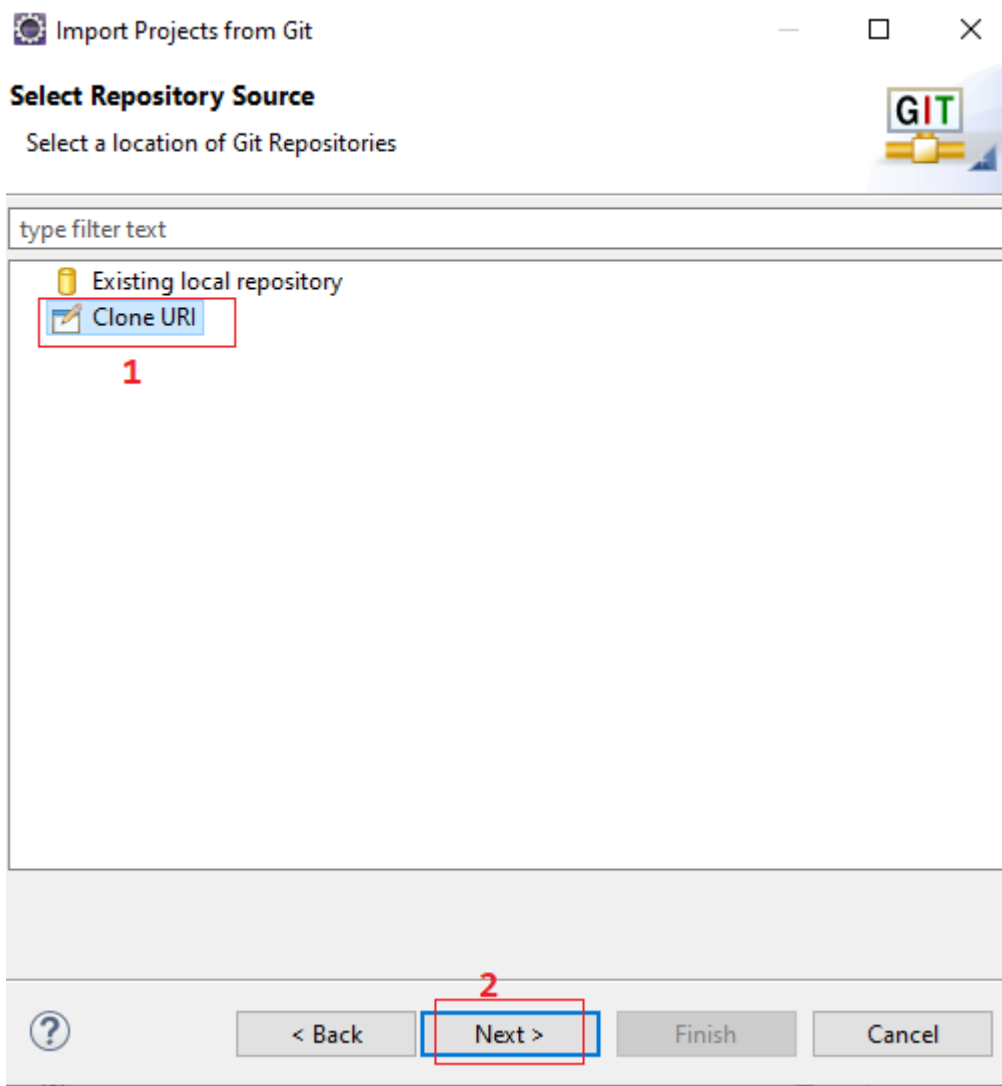
- Ajouter ensuite dans le fichier .gitignore la ligne *.unwanted.
- Commiter ensuite les modifications.

d. Cloner un repository existant

- Pour cloner un repository, cliquer à droite de la souris sur Import. La fenêtre suivante s'affiche :



1. Sélectionner « Projets from Git ».
2. Cliquer sur Next >. La fenêtre suivante s'affiche :



1. Sélectionner « Clone URI ».
2. Cliquer sur Next >. La fenêtre suivante s'affiche :

Import Projects from Git

Source Git Repository
Enter the location of the source repository.

Location

URI:

Host:

Repository path:

Connection

Protocol:

Port:

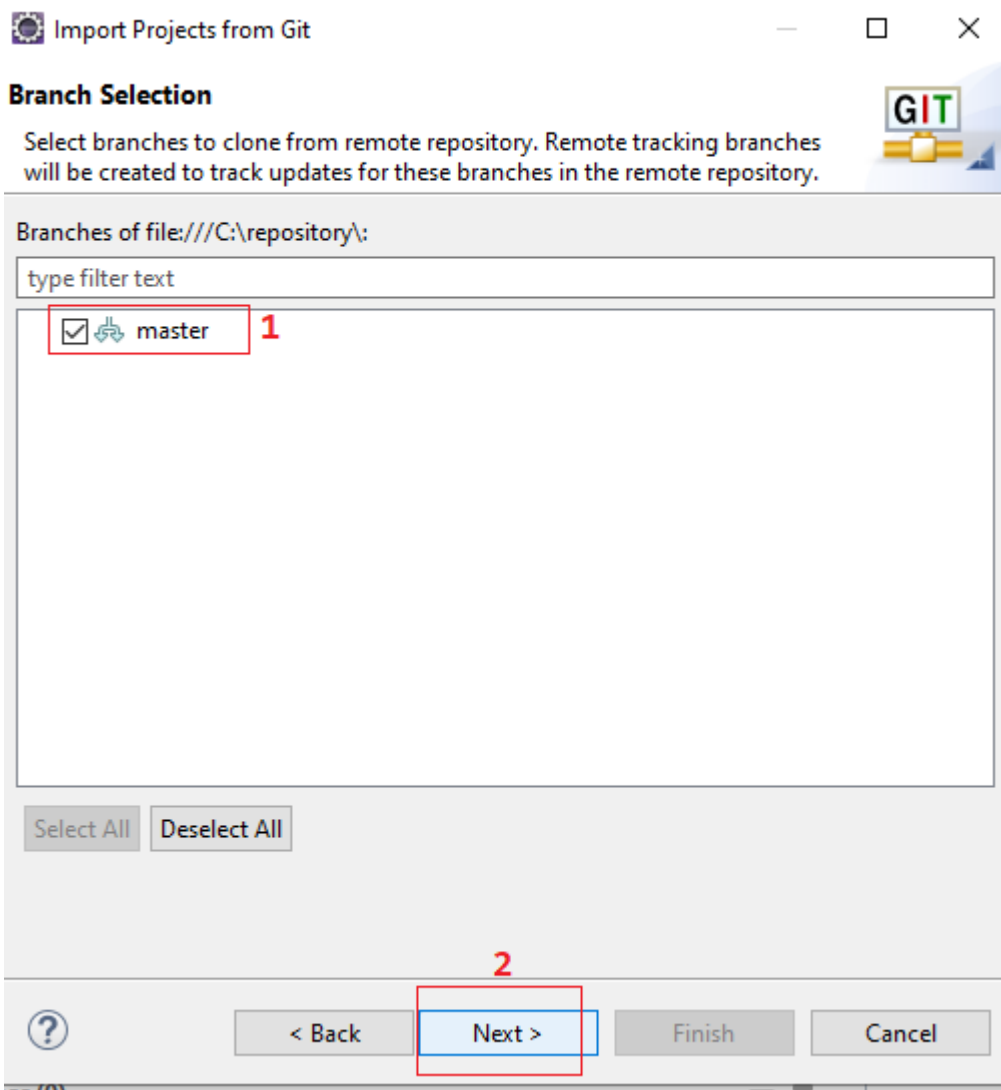
Authentication

User:

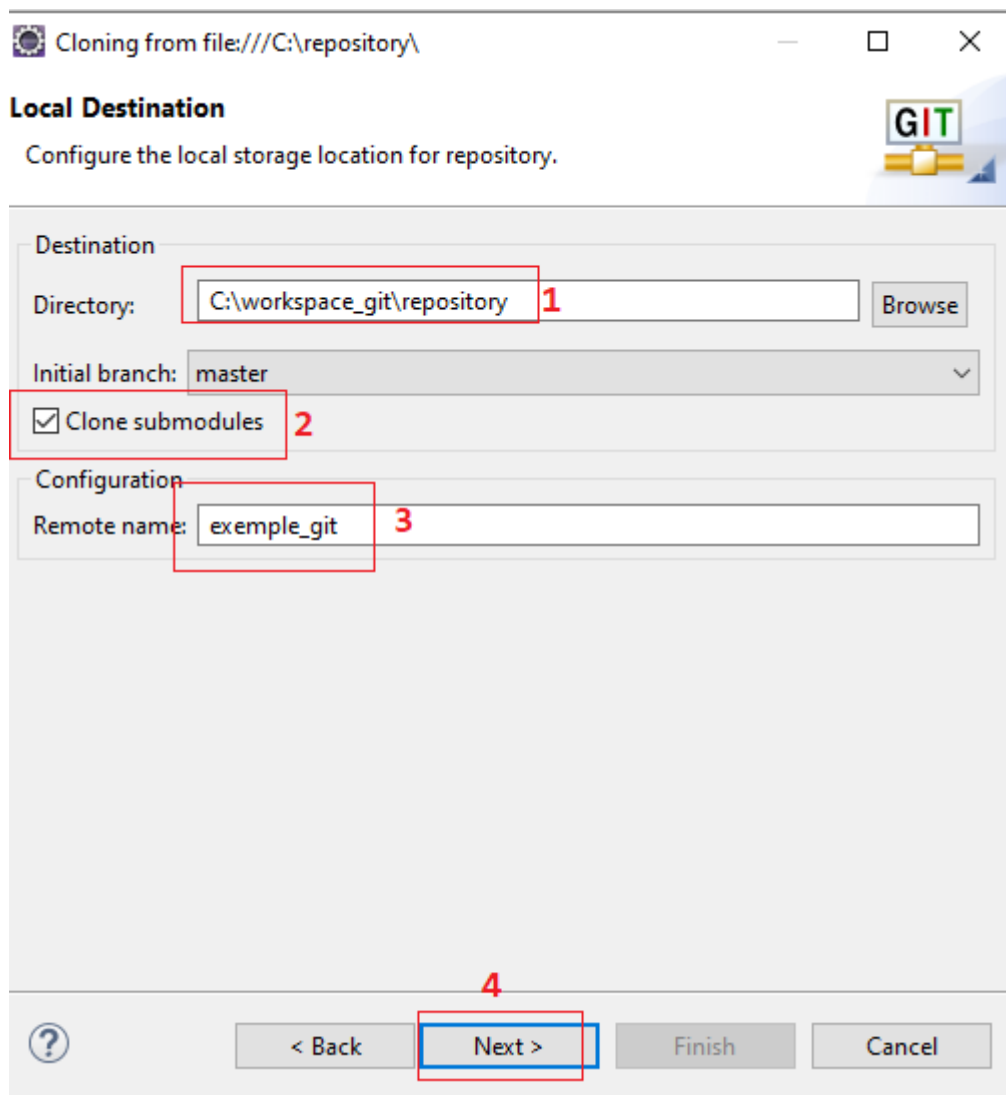
Password:

☐ Store in Secure Store

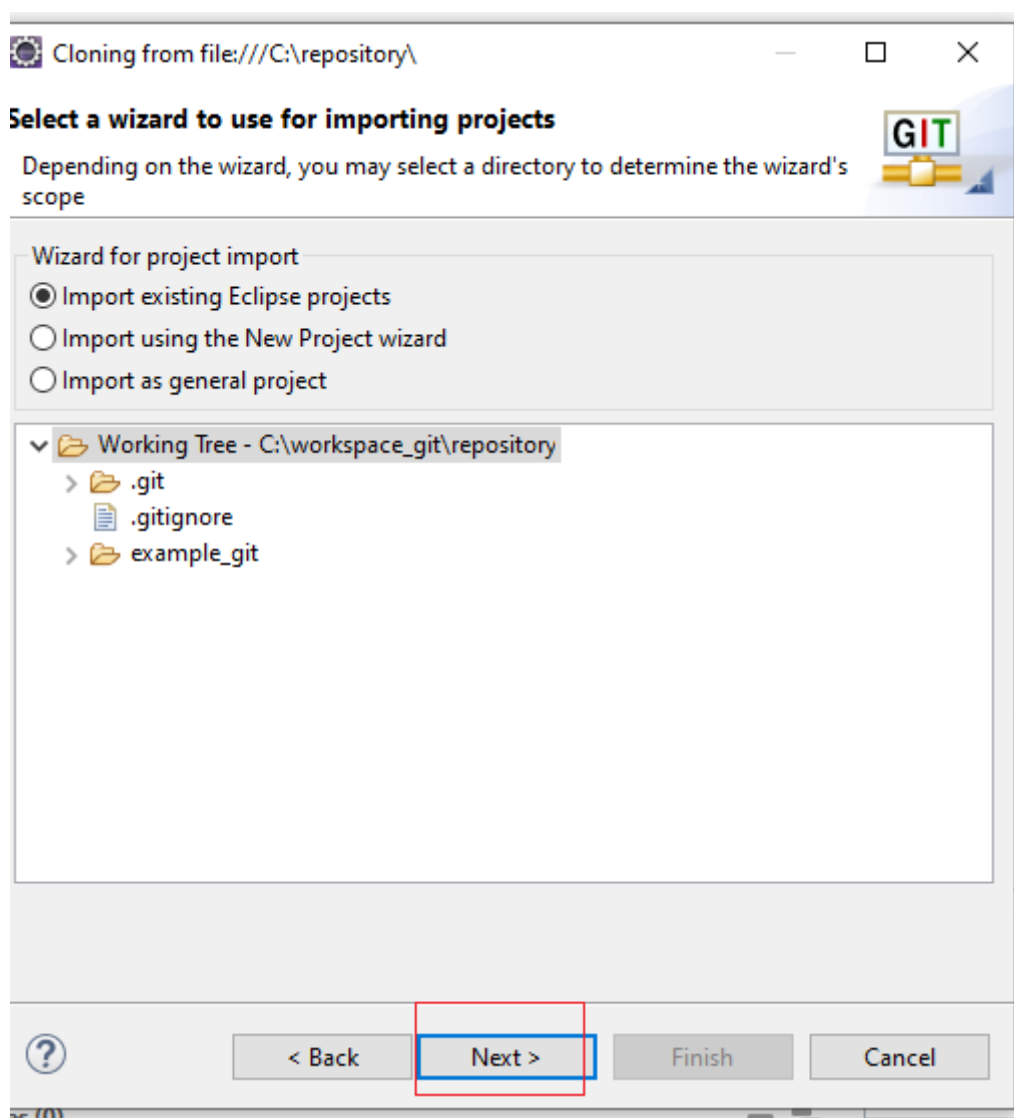
1. Entrer l'URI de votre repository (ici il s'agit d'un repository local).
2. Choisir file (pour un repository remote, faut choisir le protocole git).
3. Cliquer sur Next >. La fenêtre suivante s'affiche :



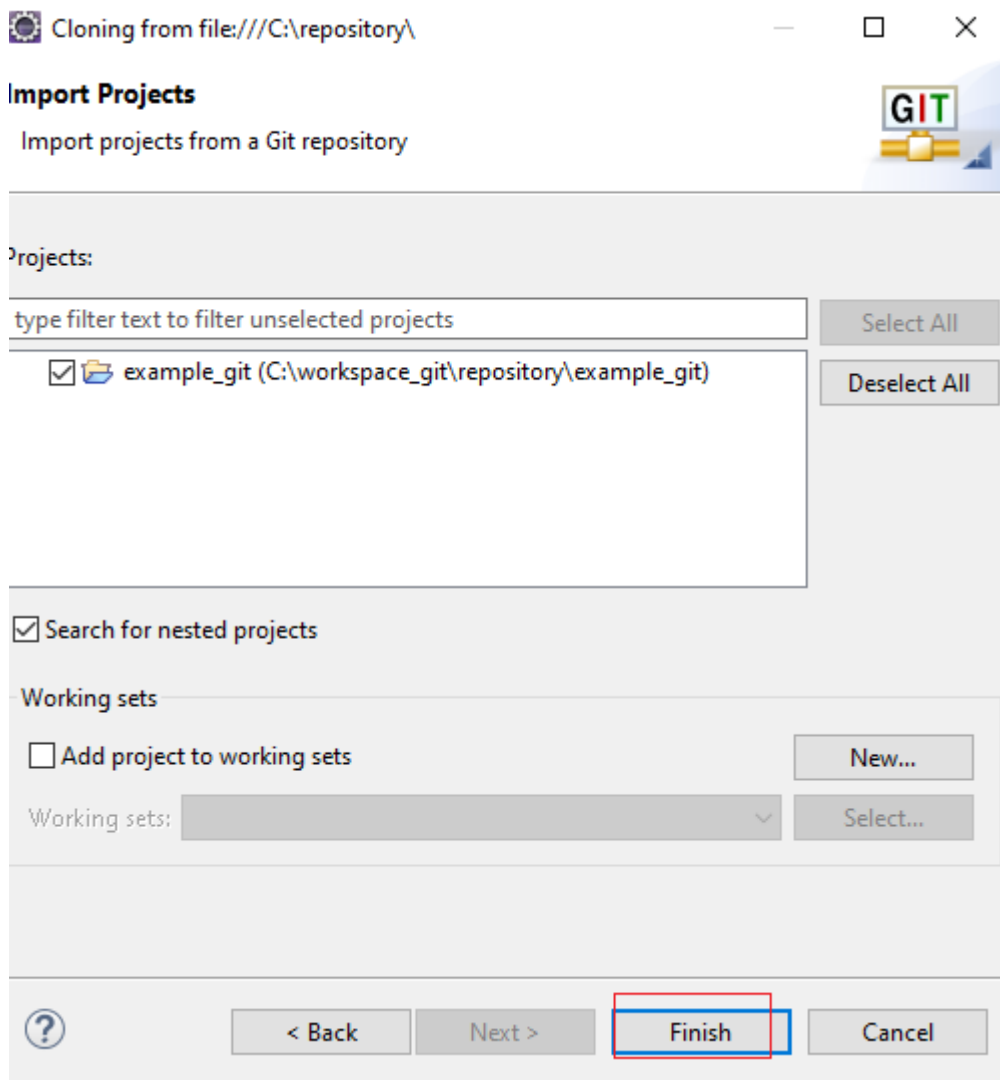
1. Sélectionner la branche (ici il y a une seule branche).
2. Cliquer sur Next >. La fenêtre suivante s'affiche :



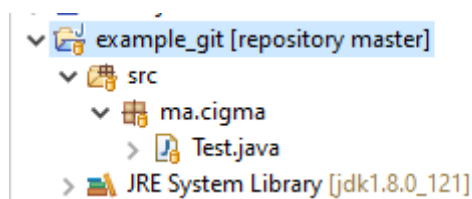
1. Entrer le chemin de votre dossier dans lequel le repository sera importé.
2. Cacher « Clone submodules ».
3. Entrer le nom que vous voulez donner au repository importé.
4. Cliquer sur Next >. La fenêtre suivante s'affiche :



1. Cliquer sur Next >. La fenêtre suivante s'affiche :



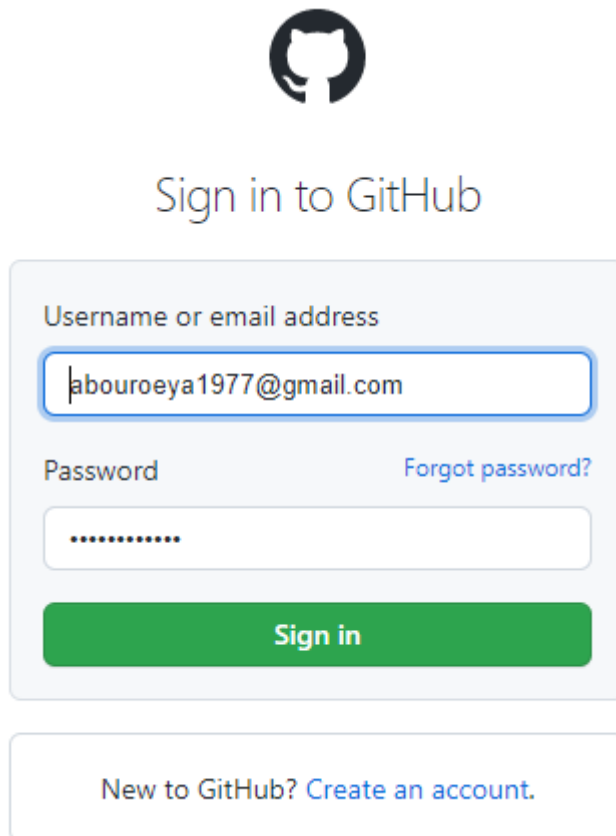
1. Cliquer sur Finish et vérifier que le projet a été bien importé au niveau d'Eclipse :



e. Utiliser un Repository Remote (exemple GITHUB)

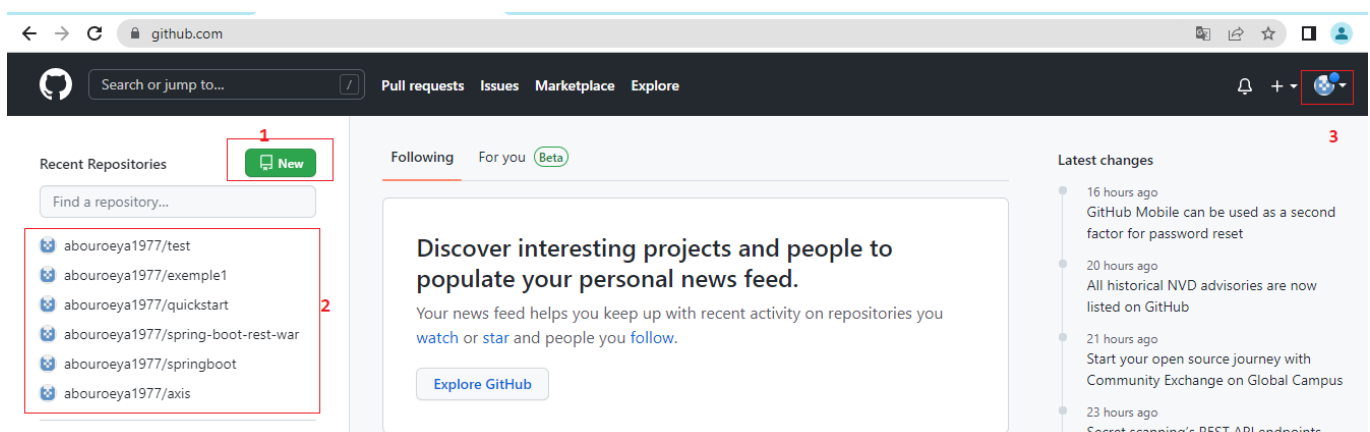
1. Création d'un compte GITHUB et créer un nouveau Repository

- Commencer par créer un compte au niveau Github. Le lien de Github est <https://github.com/>
- S'authentifier avec votre compte Github :



The image shows the GitHub sign-in page. At the top is the GitHub logo and the text "Sign in to GitHub". Below this is a form with two input fields: "Username or email address" containing "abouroeya1977@gmail.com" and "Password" with masked characters. A "Forgot password?" link is next to the password field. A green "Sign in" button is below the fields. At the bottom, a box contains the text "New to GitHub? [Create an account.](#)".

La fenêtre suivante s'affiche :



1. Cliquer sur New pour créer un Repository. La fenêtre suivante s'affiche :

Create a new repository

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

Owner *

 abouroeya1977 ▾

Repository name *

exemple ✓

1

Great repository names are short and unique. [example is available.](#) Need inspiration? How about [expert-octo-goggles?](#)

Description (optional)

☒  Public

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

2

☐  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 brief description for your project. [Learn more.](#)

- Entrer le nom de votre repository (ici : exemple).
- Cocher Public.
- Cliquer ensuite sur « Create Repository ». La fenêtre suivante s'affiche :

Quick setup — if you've done this kind of thing before 1

 Set up in Desktop

or

HTTPS SSH

<https://github.com/abouroeya1977/exemple.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

- Observer le lien de votre Repository.
2. La liste de vos Repositories s'affiche ici.
 3. Cliquer sur l'icône pour configurer la sécurité. La fenêtre suivante s'affiche :

Signed in as
abouroeya1977

😊 Set status

Your profile

Your repositories

Your codespaces

Your projects

Your stars

Your gists

Upgrade

Feature preview

Help

Settings

Sign out

1. Cliquer sur Settings. La fenêtre suivante s'affiche :

Security

- 🛡️ Code security and analysis

Integrations

- 🔗 Applications
- 🕒 Scheduled reminders

Archives

- 📁 Security log
- 📁 Sponsorship log

<> Developer settings

Company

You can @mention your company's GitHub organization to link it.

Location

All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our [privacy statement](#) to learn more about how we use this information.

Update profile

Contributions & Activity

1. Cliquer sur « **Developer settings** ». La fenêtre suivante s'affiche :

[Settings](#) / Developer settings

🔗 GitHub Apps

🔗 OAuth Apps

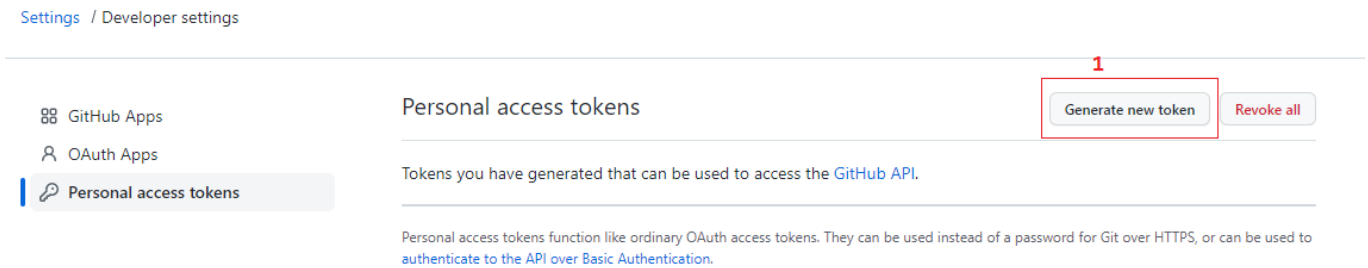
🔗 Personal access tokens

GitHub Apps

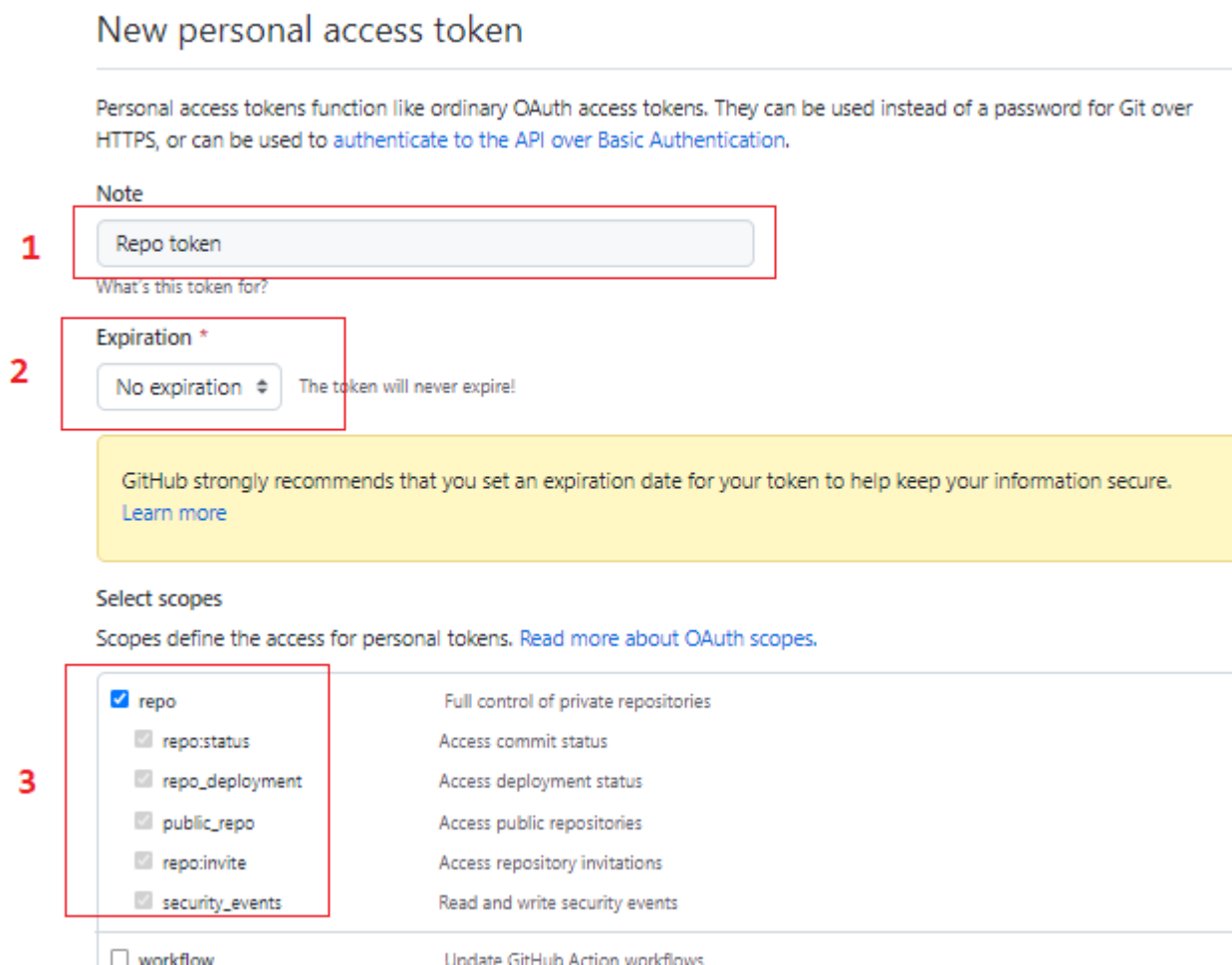
New GitHub App

Want to build something that integrates with and extends GitHub? [Register a new GitHub App](#) to get started developing on the GitHub API. You can also read more about building GitHub Apps in our [developer documentation](#).

1. Cliquer sur « Personal access tokens ». La fenêtre suivante s'affiche :



1. Cliquer sur le bouton « Generate new token ». La fenêtre suivante s'affiche :




1. Entrer une description de votre token.
2. Préciser le délai de validité de votre token (ici Token ouvert).
3. Cocher repo.
4. Cliquer ensuite sur le bouton en bas « **Generate Token** ». Vérifier que le Token a été bien généré comme le montre l'écran suivant :

Personal access tokens

[Generate new token](#)[Revoke all](#)

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

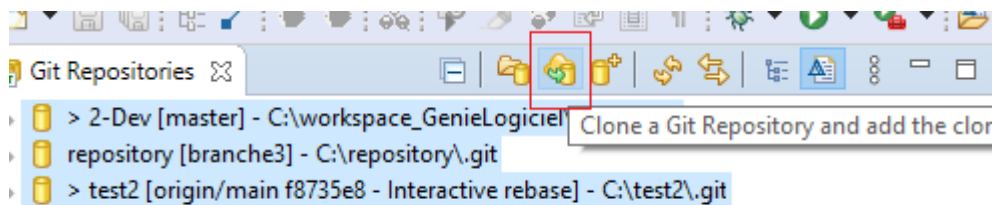
✓ ghp_wnJgx1X2G3E01FqSLRUpWUuwGye71q0mHMM2 

[Delete](#)

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

2. Cloner le Repository au niveau d'Eclipse

- Au niveau d'Eclipse, ouvrir la perspective Git.
- Au niveau de la vue **Git Repository**, cliquer sur l'icône ci-dessous :



La fenêtre suivante s'affiche :

Clone Git Repository

Source Git Repository

Enter the location of the source repository.

Location

URI: Local File...

Host:

Repository path:

Connection

Protocol: ▼

Port:

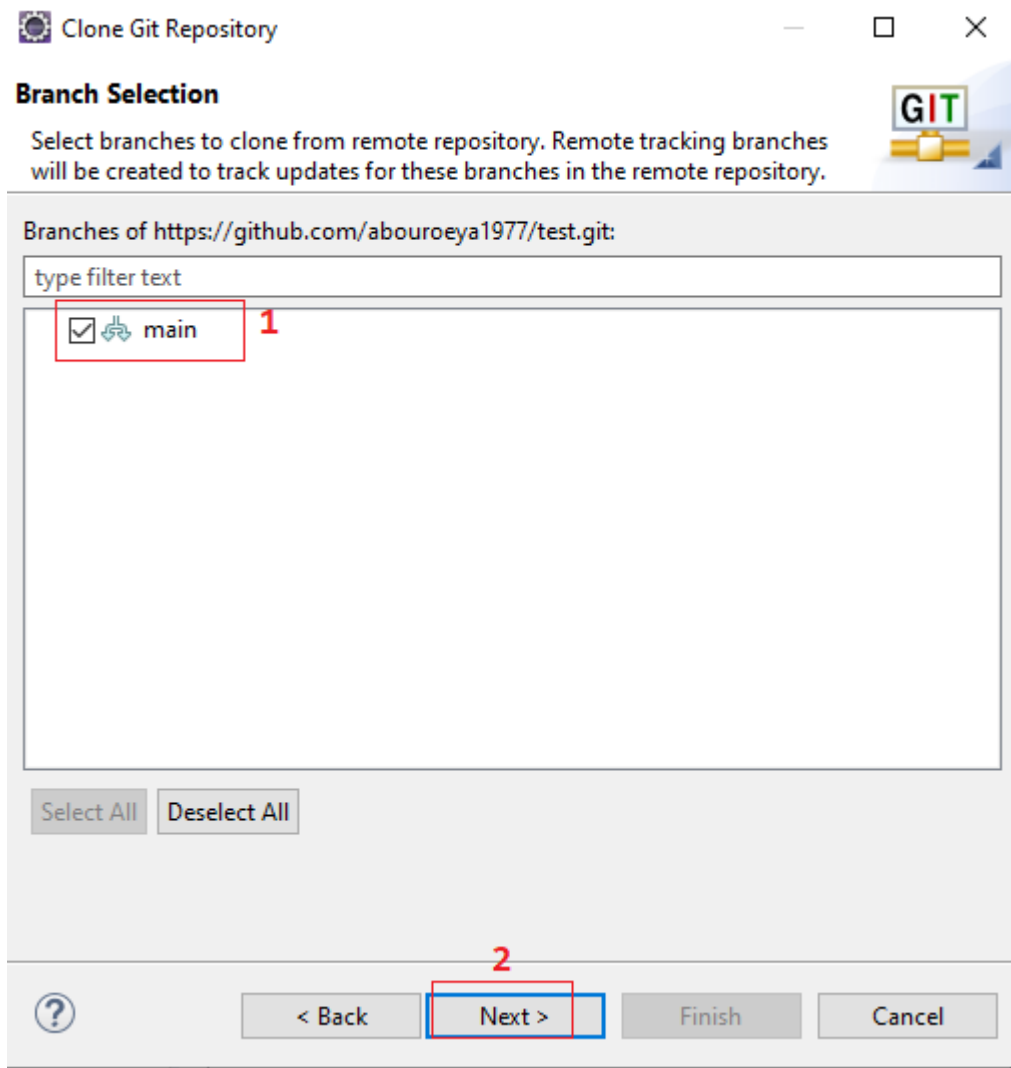
Authentication

User:

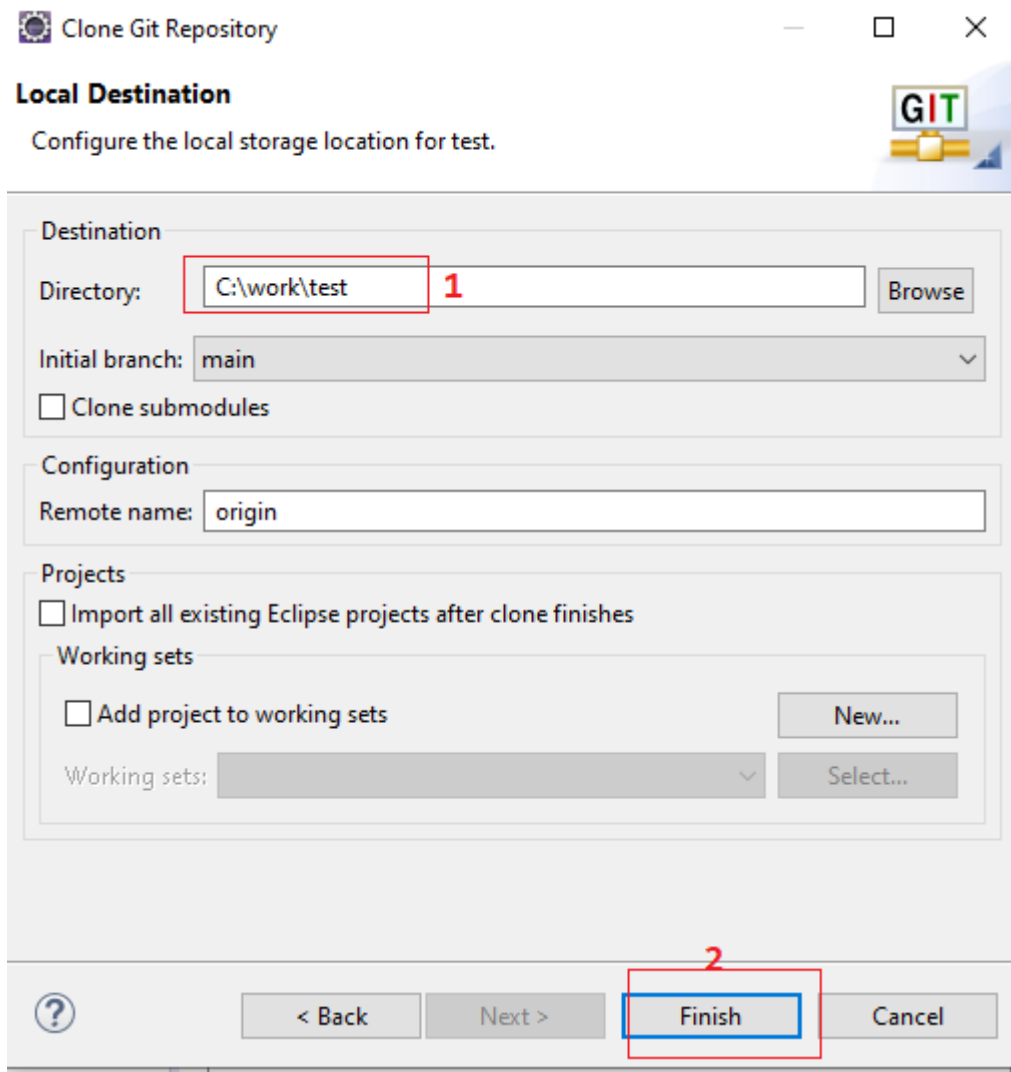
Password:

☒ Store in Secure Store

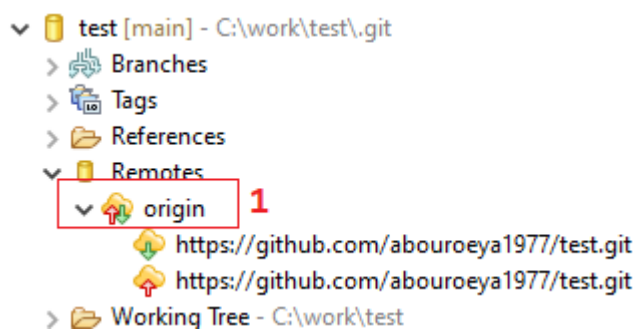
1. Entrer le lien de votre Repository.
2. Cliquer sur **Next >**. La fenêtre suivante s'affiche :



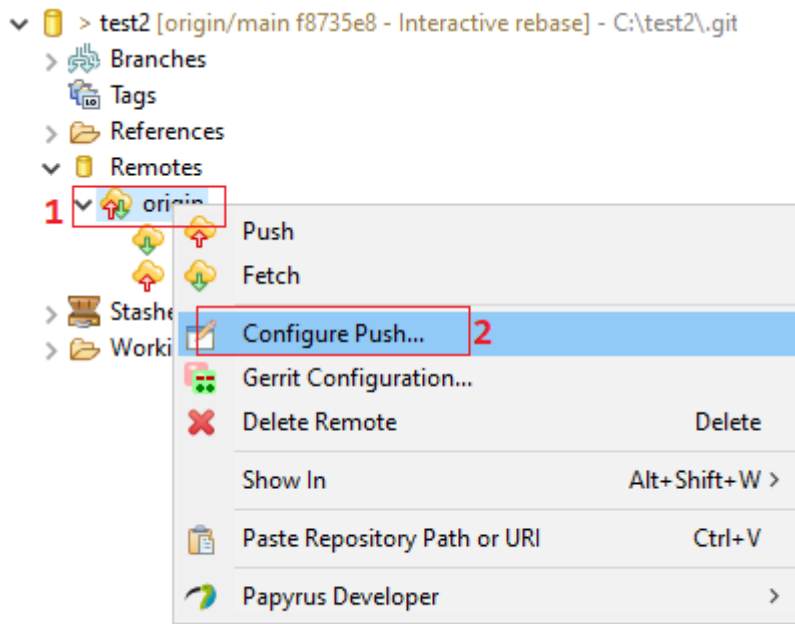
1. Choisir votre branche.
2. Cliquer sur **Next >**. La fenêtre suivante s'affiche :



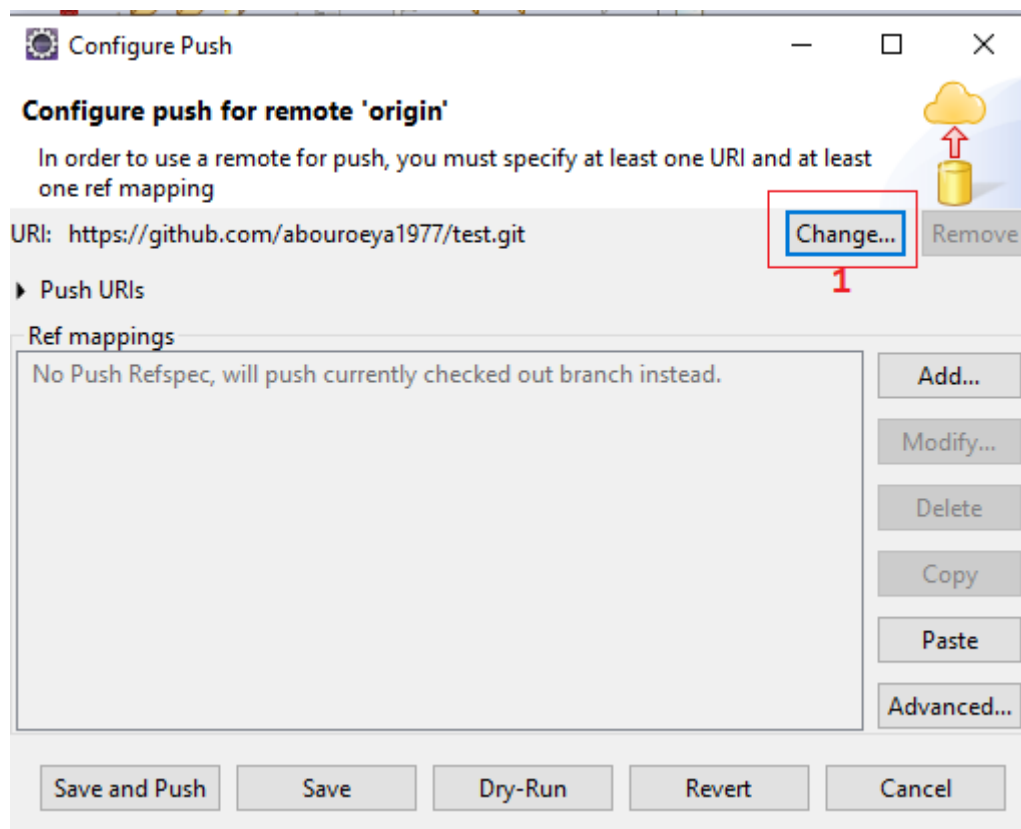
1. Entrer le dossier dans lequel sera placé votre clone.
2. Cliquer sur **Finish**. Vérifier que le repository a été cloné et ajouté au niveau de votre vue comme le montre l'écran suivant :



1. Cliquer à droite sur votre branche. La fenêtre suivante s'affiche :



2. Cliquer sur « **Configure Push...** ». La fenêtre suivante s'affiche :




1. Cliquer sur le bouton **Change...** La fenêtre suivante s'affiche :

Select a URI

Source Git Repository

Enter the location of the source repository.



Location

URI:

Host:

Repository path:

Connection

Protocol:

Port:

Authentication

User:

Password:

☒ Store in Secure Store

1. Entrez le token que vous avez généré dans GITHUB.

2. Cliquez sur le bouton **Finish**.

1. Entrez le token que vous avez généré dans GITHUB.
2. Cliquez sur le bouton **Finish**.