

D-CLIC, FORMEZ-VOUS AU NUMÉRIQUE AVEC L'OIF

« Programme D-CLIC de l'Organisation internationale de la Francophonie »

Atelier Git et GitHub pour Débutants

Objectifs de l'atelier

- Comprendre ce qu'est Git et son importance dans le développement.
- Apprendre à utiliser les commandes de base de Git.
- Créer et configurer un compte GitHub.
- Gérer un projet en ligne avec GitHub.
- Adopter de bonnes pratiques pour collaborer efficacement.

1. Introduction à Git et GitHub

Qu'est-ce que Git ?

Git est un système de contrôle de version qui permet :

- De suivre les modifications dans le code source.
- De travailler en équipe sur le même projet sans conflits.
- De revenir à une version précédente si nécessaire.

Qu'est-ce que GitHub ?

GitHub est une plateforme en ligne qui permet :

- D'héberger des projets utilisant Git.
- De collaborer avec d'autres développeurs à travers des outils comme les pull requests et les issues.
- De partager du code publiquement ou privément.

Formations aux métiers du numérique avec l'OIF en :
Développement Web - Développement mobile - Marketing numérique
Culture numérique - Cybersécurité - Intelligence Artificielle
Cyber Academy - Gouvernance du numérique
(+228) 719 987 05 dclicoiftogo@gmail.com



« Programme D-CLIC
de
l'Organisation
internationale
de la Francophonie »

ORGANISATION
INTERNATIONALE DE
la francophonie

D-CLIC
FORMEZ-VOUS AU NUMÉRIQUE !

2. Installation et Configuration de Git

Installation de Git

- **Windows :**
 1. Téléchargez Git depuis: <https://git-scm.com/download/win>
 2. Installez-le en suivant les instructions de l'assistant d'installation.
 3. Vérifiez l'installation en ouvrant le terminal (Git Bash ou Command Prompt avec le raccourci clavier **Win+R** et tapez **cmd** suivis de la touche **Entrer**) et tapez : **git --version**
 - **Mac :**
 1. Installez Homebrew (si ce n'est pas déjà fait) :
/bin/bash -c "\$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh>)"
 2. Installez Git via Homebrew :
brew install git
 3. Vérifiez l'installation :
git --version
 - **Linux :**
 1. ouvrez le terminal et tapez la commande suivante pour installer Git :
sudo apt-get install git
 2. Vérifiez l'installation :
git --version
-

3. Création d'un Compte GitHub

1. Rendez-vous sur <https://github.com>.
2. Cliquez sur **Sign Up**.
3. Renseignez :
 - Votre adresse email.
 - Un mot de passe.
 - Un nom d'utilisateur.
4. Suivez les instructions pour valider votre compte.
5. Connectez-vous à votre compte GitHub.

Configuration de Git (pour tous les systèmes d'exploitation)

1. Après installation, configurez Git avec vos informations :

```
git config --global user.name "Votre Nom"  
git config --global user.email "votre.email@example.com"
```

2. Vérifiez la configuration :

```
git config --list
```

4. Commandes de Base Git pour système d'Exploitation (Windows; linux; Mac)

1. Initialiser un Repository Local

Créer un nouveau dossier

```
mkdir mon-projet
```

Aller dans le dossier

```
cd mon-projet
```

Initialiser Git

```
git init
```

Créer un fichier README.md

```
echo. > README.md #commande pour windows au niveau de l'invite de commande
```

Créer un fichier README.md

```
touch README.md #commande pour Linux et MacOS
```

Formations aux métiers du numérique avec l'OIF en :
Développement Web – Développement mobile – Marketing numérique
Culture numérique – Cybersécurité – Intelligence Artificielle
Cyber Academy – Gouvernance du numérique
(+228) 719 987 05 dclicoiftogo@gmail.com

2. Ajouter et Valider des Fichiers

`echo "Description du projet" > README.md`

Ajoutez un fichier au suivi :

`git add README.md`

Validez les modifications :

`git commit -m "Premier commit"`

3. Vérifier l'état et l'historique

Vérifiez l'état des fichiers :

`git status`

Affichez l'historique des commits :

`git log`

5. Utilisation de GitHub

Créer un Repository en Ligne

1. Connectez-vous à GitHub.
2. Cliquez sur **New Repository**.
3. Donnez un nom au repository (ex : "mon-projet").
4. Choisissez les options de visibilité (public ou privé).
5. Cliquez sur **Create Repository**.

Lier le Repository Local à GitHub

Dans le terminal, liez votre repository local :

`git remote add origin https://github.com/votre-nom-utilisateur/mon-projet.git`

`git branch -M main`

`git push -u origin main`

Formations aux métiers du numérique avec l'OIF en :
Développement Web – Développement mobile – Marketing numérique
Culture numérique – Cybersécurité – Intelligence Artificielle
Cyber Academy – Gouvernance du numérique
(+228) 719 987 05 dclicoiftogo@gmail.com

Mettre à Jour le Repository

Ajoutez et poussez les modifications :

```
git add .  
git commit -m "Ajout d'un nouveau fichier"  
git push origin main
```

6. Collaboration avec GitHub

Cloner un Repository existant

Pour travailler sur un repository existant :

```
git clone https://github.com/devBonkey/atelier-git-github-dclic-togo.git
```

Créer une Branche

```
git branch nouvelle-fonctionnalite  
git checkout nouvelle-fonctionnalite
```

NB : Dans notre cas, pour le test, au lieu de "nouvelle-fonctionnalité", il faudra mettre "prénom-nom".

Ajouter le fichier "résumé.docx"

- Créez un fichier nommé **résumé.docx** dans le dossier cloné.
- Ajoutez-y le résumé de ce que vous avez retenu lors de l'atelier Git et Github..

Ensuite, ajoutez et validez ce fichier avec les commandes suivantes :

```
git add résumé.docx
```

```
git commit -m "Ajout du fichier résumé.docx"
```

Formations aux métiers du numérique avec l'OIF en :
Développement Web – Développement mobile – Marketing numérique
Culture numérique – Cybersécurité – Intelligence Artificielle
Cyber Academy – Gouvernance du numérique
(+228) 719 987 05 dclicoiftogo@gmail.com



Faire un Pull Request

1. Poussez votre branche sur GitHub :
git push origin nouvelle-fonctionnalite

NB : Dans notre cas, pour le test, au lieu de "nouvelle-fonctionnalité", il faudra mettre "prénom-nom".

2. Allez sur GitHub et créez une **pull request** depuis votre branche **prenom-nom** vers la branche principale (**main**).

7. Bonnes Pratiques sur GitHub

- **Nommer clairement les commits :**
 - Mauvais : **fix**
 - Bon : **Correction du bug d'affichage sur la page d'accueil**
- **Utiliser des branches :**
 - Créez une branche par fonctionnalité.
- **Faire des pull requests :**
 - Ajoutez des descriptions claires et des commentaires si nécessaire.
- **Projets publics ou privés :**
 - Ajoutez un fichier README.md pour expliquer le projet.
 - Incluez un fichier LICENSE si besoin.

8. Exercices Pratiques

NB : Afin de pouvoir accéder à vos repositories sans demande de permission et les corriger, il faudra rendre tous les dépôts publics.

Exercice 1 : Initialisation et Premier Commit

1. Créez un dossier local.
2. Initialiser un repository Git.
3. Ajoutez un fichier README.md avec une description du projet.
4. Commitez les modifications.

Exercice 2 : Collaboration

1. Créez un repository sur GitHub.
2. Cloner le repository d'un camarade.

Formations aux métiers du numérique avec l'OIF en :
Développement Web – Développement mobile – Marketing numérique
Culture numérique – Cybersécurité – Intelligence Artificielle
Cyber Academy – Gouvernance du numérique
(+228) 719 987 05 dclicoiftogo@gmail.com



« Programme D-CLIC
de
l'Organisation
internationale
de la Francophonie »

ORGANISATION
INTERNATIONALE DE
la francophonie

D-CLIC
FORMEZ-VOUS AU NUMÉRIQUE !

3. Ajoutez un fichier et faites une pull request.

Exercice 3 : Gestion d'un Projet Local avec Git (Gérez un projet local en utilisant Git.)

Réaliser les tâches suivant :

1. Créez un dossier nommé **mon-projet-dclic**.
2. Initialiser un repository Git dans ce dossier.
3. Créez un fichier Word nommé **rapport.docx** contenant une liste de vos objectifs pour la formation.
4. Ajoutez le fichier au suivi Git et validez les modifications avec un message clair.
5. Modifiez le fichier **rapport.docx** en ajoutant deux nouveaux objectifs.
6. Validez cette modification et vérifiez l'historique des commits.

Exercice 4 :Collaboration sur un Projet avec GitHub

Simulez une collaboration en ligne avec Git et GitHub en utilisant un fichier Word. Voici les tâches :

1. Créez un repository en ligne nommé **projet-collaboration** sur GitHub.
2. Liez un repository Git local à ce repository en ligne.
3. Travaillez sur une nouvelle branche nommée **nouveau-contenu**.
4. Ajoutez un fichier Word nommé **plan.docx** contenant une idée d'organisation pour un projet fictif.
5. Proposez vos modifications en créant un pull request sur GitHub.
6. Fusionnez vos modifications dans la branche principale et synchronisez les changements avec votre repository local.

Les éléments à rendre :

- Capture d'écran de votre invite de commande ou terminal après avoir tapé la commande "**git config --list**"
- Votre nom d'utilisateur github
- Le lien vers le repository de chaque exercice.

