

## Durée : 2 jours

Réf : Perf-Git

## Perfectionnement Git

Cette formation permet d'approfondir l'utilisation de GIT pour gérer plus efficacement des codes source : gestion des branches, gestion des sous modules, réécriture de l'historique, ...

## Objectifs

- Gérer les branches des projets
- Personnaliser le comportement de Git
- Connaitre les Workflow classiques
- Gérer les sous modules et les sous arbres
- Modifier l'historique

## Public concerné

- Développeur, Chef de projet

## Pré requis

- Utilisation de Git

## Evaluation des acquis

- TP et QCM pendant la formation

## Méthodes et moyens pédagogiques

- Travaux pratiques
- Pédagogie active basée sur des exemples, des démonstrations, des partages d'expériences, des cas pratiques et une évaluation des acquis tout au long de la formation
- Un poste de travail par personne
- Support électronique

## 1 Rappels

Présentation de GIT

Le répertoire GIT et le répertoire de travail.

L'index ou staging area.

Les concepts de branche, tag et de dépôt.

Gestion locale des fichiers

Gestion des branches

Dépôts distants

## 2 Améliorer la gestion des branches

Mémoriser l'état de modifications (git stash)

Utilisation des worktree

Notion de commit Fast-Forward

Notion de commit squash

Intérêt et utilisation de rebase

Intérêt et utilisation de cherry-pick

## 3 Présentation de workflow Git classiques

Trunk

Branches de fonctionnalités

GitFlow

Fork

## 4 Hook

Évènements

Script

## 5 Personnalisation avec .gitattributes

Principe

Utilisation de filtres standard

Écriture

## 6 Stockage de fichiers binaires volumineux avec Git LFS

Principe

Installation

Mise en oeuvre

## 7 Sous Modules

Cas d'utilisation

Création

Gestion

## 8 Sous Branches

Cas d'utilisation

Création

Gestion

## Points forts

- Groupe de 4 personnes maximum
- Alternance théorie/pratique
- Assistance après la formation

## Contact

- 04 58 00 02 22
- [contact@webformation.fr](mailto:contact@webformation.fr)

## Plus d'infos sur

<https://www.webformation.fr>

## 9 Recherche de bug

Recherche dans les logs  
Bissect

## 10 Patch

Intérêts  
Création d'un patch  
Application d'un patch

## 11 Réécriture de l'historique

Rebase interactif  
Filter-branch  
Filter-repo