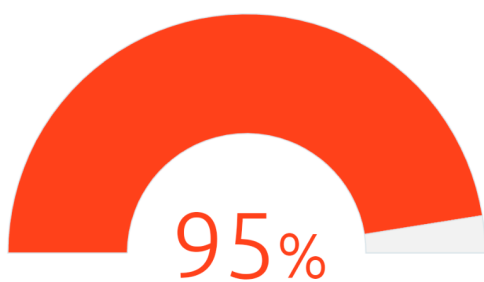
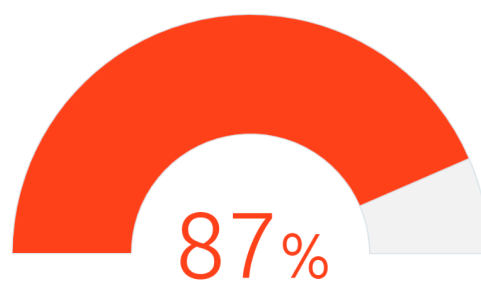


PROGRAMME DE FORMATION

Versionner et collaborer avec Git



ÉVALUATION ÉQUIPE PÉDAGOGIQUE



CONTENU & ORGANISATION

OBJECTIFS

Apprenez à tirer parti de toute la puissance de Git et de ses fonctionnalités cachées pour un meilleur travail collaboratif.

À l'issue de la formation, le participant ou la participante sera en mesure de :

- Versionner son travail
- Collaborer et résoudre des conflits de modifications
- Mettre en place un flux de travail pérenne
- Comprendre les mécanismes internes de Git

PRÉ-REQUIS

- Ordinateur portable à apporter
- Git installé et à jour

PUBLIC VISÉ

Développeuses et développeurs.

DURÉE

2 jours.

MOYENS ET MÉTHODES PÉDAGOGIQUES

- Présentation des concepts théoriques.
- Ateliers avec cas pratiques.

MODALITÉS D'ÉVALUATION

- Diagnostic préalable à l'inscription
- Evaluation en contrôle continue (ateliers et exercices pendant la formation)
- QCM en fin de formation

PROGRAMME

Introduction

- A quoi sert le versionning
- Les alternatives
- Présentation de Git
- Les outils disponibles

Configuration

- Configuration de base
- Configuration des préférences
- Configuration des autorisations
- Signature des commits
- Omission de fichiers et répertoires

Les bases

- Initialiser un projet
- Gérer les dépôts
- Mettre à jour les références
- Récupérer des modifications
- Publier des modifications
- Les commits
- Enregistrer un commit
- Les entrailles de Git
 - ◆ Branches
 - ◆ Tags
 - ◆ Commits
 - ◆ Tree
 - ◆ Blob

Premiers pas

- Cycle de vie d'un fichier

- Vérifier l'état d'un fichier
- Vérifier les modifications
- Indexer les modifications
- Enregistrer les modifications
- Visionner les détails d'un commit
- Exercices

Les branches

- Créer une branche
- Naviguer entre les branches
- Fusionner des branches
 - ◆ Fast-Forward
 - ◆ Fusion à 3 sources
 - ◆ Conflits de fusion
- Exercices

L'historique

- Visionner l'historique d'une branche
- Chercher la présence d'un commit dans une branche
- Chercher la personne responsable d'une modification
- Chercher le commit responsable d'une modification
- Rejouer un commit
- Ré-écrire l'historique
- Ré-écrire l'historique manuellement
- Naviguer dans l'historique
- L'historique caché
- Manipuler le cycle de vie d'un fichier
- Annuler les modifications d'un commit
- Sauvegarde temporaire
- Exercices

Gestion de projet

- Interfaces web
- Pull/Merge Requests

→ Organisation des branches

- ◆ Git flow
- ◆ GitHub flow
- ◆ GitLab flow

Bonnes pratiques

- Commits atomiques
- Commits sémantiques
- Décrire un commit
- Relecture d'une Pull/Merge Request

NOTES

Nous pouvons ajuster la formation à vos besoins, en mettant l'accent sur un aspect plus particulier du développement.

PSH

Les personnes en situation de handicap sont invitées à contacter en toute confidentialité notre équipe pour définir les modalités d'accueil de la formation PSH@les-tilleuls.coop.