

# Manuel Git Formation

## Manuel d'utilisation : Partage des fichiers du cours avec Git & GitHub

### Introduction

Ce manuel explique comment :

- le **formateur** envoie les documents du cours sur GitHub,
- les **participants** récupèrent uniquement les fichiers modifiés, et non tout le projet.

### 1. Fonctionnement général

- Git suit les versions des fichiers.
- GitHub héberge le projet en ligne.
- Le formateur pousse (push) les mises à jour.
- Les participants tirent (pull) les nouveautés.

### 2. Pré-requis

#### Pour tous

- Installer Git : <https://git-scm.com>

#### Pour le formateur

- Compte GitHub obligatoire.

## 3. Procédure : Formateur

### A. Initialiser le projet local

Dans le dossier du cours (C:/ ReactProjects) :

Copier le fichier readme.md qui se trouve dans le dossier ressource vers le dossier ReactProjects

```
git init
```

Ajouter tous les fichiers :

```
git add .
```

Créer un premier commit :

```
git commit -m "Initialisation du projet"
```

## B. Créer le dépôt sur GitHub

- Créer un **nouveau repository** vide.
- Copier l'URL HTTPS.

Associer le dossier local au dépôt GitHub :

```
git remote add origin https://github.com/kbma/ReactProjects.git
```

Renommer la branche principale :

```
git branch -M main
```

Envoyer le projet :

```
git push -u origin main
```

## C. Envoyer les mises à jour à chaque séance

Après modification des fichiers :

1. Ajouter les fichiers modifiés

```
git add .
```

2. Valider les changements

```
git commit -m "Mise à jour"
```

3. Envoyer vers GitHub

```
git push
```

## 4. Procédure : Participants

### A. Télécharger le cours (une seule fois)

Aller sur le disque local C:/

Lancer cette commande

```
git clone https://github.com/kbma/ReactProjects.git
```

Cette commande télécharge tout le projet une seule fois.

## B. Récupérer les nouvelles mises à jour

À chaque mise à jour, pour avoir uniquement les fichiers modifiés.

Allez dans le dossier **C:/ ReactProjects** et lancer cette commande

```
git pull
```

Git télécharge **seulement les différences**, pas tout le dossier.

## 5. Commandes essentielles

### Formateur

```
git init
git add .
git commit -m "message"
git remote add origin https://github.com/kbma/ReactProjects.git
git push
```

### Participants

```
git clone https://github.com/kbma/ReactProjects.git
git pull
```

## 6. Organisation recommandée

- Un dossier par chapitre :

01-introduction

02-react-debut

03-api

04-exercices

- Toujours annoncer quand une mise à jour est disponible.

## 7. Bonnes pratiques

- Messages de commit clairs.

- Structure simple.
- Ne jamais modifier les fichiers du participant directement.