

## Formation Git : TP

El Hadji Gaye

---

**Auteur** El Hadji Gaye

**Pour** Formations

**Date** 16/09/2024

---

**Objet** Formation Git : TP

---

<b>I)</b>	<b>Préambule sur les travaux pratiques</b> .....	3
<b>II)</b>	<b>Le projet MyWebSite</b> .....	4
<b>III)</b>	<b>TP avec ECommerceSiteMerge</b> .....	5
1.	Créer le dossier Git/tp/ECommerceSiteMerge .....	5
2.	Créer la branche feature .....	6
3.	Retour sur la branche master .....	6
4.	Merge de feature sur la branche master .....	6
<b>IV)</b>	<b>TP avec ECommerceSiteRebase</b> .....	7
1.	Créer le dossier Git/tp/ECommerceSiteRebase .....	7
2.	Créer la branche feature .....	8
3.	Retour sur la branche master .....	8
4.	Rebase de feature par rapport à la branche master.....	8
5.	Merge de feature sur la branche master .....	8
<b>V)</b>	<b>TP avec ECommerceSiteMergeConflict</b> .....	9
1.	Créer le dossier Git/tp/ECommerceSiteMergeConflict .....	9
2.	Créer la branche feature .....	10
3.	Retour sur la branche master .....	10
4.	Merge de feature sur la branche master .....	10
<b>VI)</b>	<b>TP avec ECommerceSiteRebaseConflict</b> .....	11
1.	Créer le dossier Git/tp/ECommerceSiteRebaseConflict .....	11
2.	Créer la branche feature .....	12
3.	Retour sur la branche master .....	12
4.	Rebase de feature par rapport à la branche master.....	12
5.	Merge de feature sur la branche master .....	12
<b>VII)</b>	<b>TP avec ECommerceSiteGitFlow</b> .....	13
1.	Créer le dossier Git/tp/ECommerceSiteGitFlow .....	13
2.	Créer les branches pour les différents modules .....	13
3.	Mise en production : scénario 1 .....	13
4.	Mise en production : scénario 2 .....	13

## **I) Préambule sur les travaux pratiques**

Faire les différentes TP les uns à la suite des autres. N'hésitez pas à contacter le formateur en cas de besoin. Le but de ces TP est d'appliquer en pratique toutes les notions qui ont été abordé lors de la formation.

## II) Le projet MyWebSite

Créer le dépôt **MyWebSite** qui contiendra :

- Un fichier **index.html**.
- Un fichier **main.css** avec le contenu :

```
body {  
  background-color: lightblue;  
}
```

- Un fichier **main.js** avec le contenu :

```
console.log("Dans le main.js")
```

Réaliser un certain nombre de commits (par exemple une dizaine) quelconques pour vous constituer une certaine historique.

Créer au niveau du dossier **MyWebSite** un sous dossier **build** qui ne sera jamais commité.

Faire un commit implémentant un footer dans **index.html**.

5 minutes plus tard vous vous rendez compte de votre erreur ainsi vous décider de faire revert.

Le Lead Dev vous dit d'enlever votre le commit du footer et du revert dans l'historique donc proposez lui une solution.

### III) TP avec ECommerceSiteMerge

#### 1. Créer le dossier `Git/tp/ECommerceSiteMerge`

Il contiendra notre application de e-commerce.

Ouvrir **Visual Studio Code** avec le dossier **ECommerceSiteMerge**.

Créer le dépôt Git **ECommerceSiteMerge**.

Créer un fichier **index.html** avec un contenu quelconque par exemple :

```
<> index.html > html > head
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset='utf-8'>
5      <meta http-equiv='X-UA-Compatible' content='IE=edge'>
6      <title>Page Title A</title>
7      <meta name='viewport' content='width=device-width, initial-scale=1'>
8  </head>
9  <body>
10
11 </body>
12 </html>
```

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message **"A"**.

Ajouter sur le fichier **index.html** une deuxième modification quelconque et le commiter avec le message **"B"**.

Ajouter sur le fichier **index.html** une troisième modification quelconque et le commiter avec le message **"C"**.

## 2. Créer la branche *feature*

Afin d'éviter un conflit créer le fichier **index.js** avec comme contenu :

```
JS index.js
1 | You, 4 seconds ago | 1 author (You)
  | console.log("feature"); | You, 1 second ago • Uncommitted changes
```

Ajouter sur le fichier **index.js** une première modification quelconque et le commiter avec le message "1".

Ajouter sur le fichier **index.js** une deuxième modification quelconque et le commiter avec le message "2".

Ajouter sur le fichier **index.js** une troisième modification quelconque et le commiter avec le message "3".

## 3. Retour sur la branche *master*

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message "D".

## 4. Merge de *feature* sur la branche *master*

Merger la branche **feature** sur la branche **master**.

## IV) TP avec ECommerceSiteRebase

### 1. Créer le dossier `Git/tp/ECommerceSiteRebase`

Il contiendra notre application de e-commerce.

Ouvrir **Visual Studio Code** avec le dossier **ECommerceSiteRebase**.

Créer le dépôt Git **ECommerceSiteRebase**.

Créer un fichier **index.html** avec un contenu quelconque par exemple :

```
<> index.html > html > head
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset='utf-8'>
5      <meta http-equiv='X-UA-Compatible' content='IE=edge'>
6      <title>Page Title A</title>
7      <meta name='viewport' content='width=device-width, initial-scale=1'>
8  </head>
9  <body>
10
11 </body>
12 </html>
```

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message **"A"**.

Ajouter sur le fichier **index.html** une deuxième modification quelconque et le commiter avec le message **"B"**.

Ajouter sur le fichier **index.html** une troisième modification quelconque et le commiter avec le message **"C"**.

## 2. *Créer la branche feature*

Afin d'éviter un conflit créer le fichier **index.js** avec comme contenu :

```
JS index.js
1 | You, 4 seconds ago | 1 author (You)
  | console.log("feature"); | You, 1 second ago • Uncommitted changes
```

Ajouter sur le fichier **index.js** une première modification quelconque et le commiter avec le message "1".

Ajouter sur le fichier **index.js** une deuxième modification quelconque et le commiter avec le message "2".

Ajouter sur le fichier **index.js** une troisième modification quelconque et le commiter avec le message "3".

## 3. *Retour sur la branche master*

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message "D".

## 4. *Rebase de feature par rapport à la branche master*

Faire un rebase de la branche **feature** par rapport à la branche **master**.

## 5. *Merge de feature sur la branche master*

Merger la branche **feature** sur la branche **master**.

## V) TP avec ECommerceSiteMergeConflict

### 1. Créer le dossier `Git/tp/ECommerceSiteMergeConflict`

Il contiendra notre application de e-commerce.

Ouvrir **Visual Studio Code** avec le dossier **ECommerceSiteMergeConflict**.

Créer le dépôt Git **ECommerceSiteMergeConflict**.

Créer un fichier **index.html** avec un contenu quelconque par exemple :

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset='utf-8' >
5      <meta http-equiv='X-UA-Compatible' content='IE=edge' >
6      <title>Page Title</title>
7      <meta name='viewport' content='width=device-width, initial-scale=1' >
8      <script src='index.js'></script>
9  </head>
10 <body>
11     <footer>My Footer</footer>
12 </body>
13 </html>
```

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message **"A"**.

Ajouter sur le fichier **index.html** une deuxième modification quelconque et le commiter avec le message **"B"**.

Ajouter sur le fichier **index.html** une troisième modification quelconque et le commiter avec le message **"C"**.

Créer le fichier **index.js** avec comme contenu :

```
JS index.js
You, 4 seconds ago | 1 author (You)
1 console.log("feature"); | You, 1 second ago • Uncommitted changes
```

## 2. *Créer la branche feature*

Ajouter sur les fichiers **index.html** et **index.js** une première modification quelconque et le commiter avec le message "1".

Ajouter sur le fichier **index.html** et **index.js** une deuxième modification quelconque et le commiter avec le message "2".

Ajouter sur le fichier **index.html** et **index.js** une troisième modification quelconque et le commiter avec le message "3".

## 3. *Retour sur la branche master*

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message "D".

## 4. *Merge de feature sur la branche master*

Merger la branche **feature** sur la branche **master**.

## VI) TP avec ECommerceSiteRebaseConflict

### 1. Créer le dossier `Git/tp/ECommerceSiteRebaseConflict`

Il contiendra notre application de e-commerce.

Ouvrir **Visual Studio Code** avec le dossier **ECommerceSiteRebaseConflict**.

Créer le dépôt Git **ECommerceSiteRebaseConflict**.

Créer un fichier **index.html** avec un contenu quelconque par exemple :

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset='utf-8' >
5      <meta http-equiv='X-UA-Compatible' content='IE=edge' >
6      <title>Page Title</title>
7      <meta name='viewport' content='width=device-width, initial-scale=1' >
8      <script src='index.js'></script>
9  </head>
10 <body>
11     <footer>My Footer</footer>
12 </body>
13 </html>
```

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message **"A"**.

Ajouter sur le fichier **index.html** une deuxième modification quelconque et le commiter avec le message **"B"**.

Ajouter sur le fichier **index.html** une troisième modification quelconque et le commiter avec le message **"C"**.

Créer le fichier **index.js** avec comme contenu :

```
JS index.js
You, 4 seconds ago | 1 author (You)
1 console.log("feature"); | You, 1 second ago • Uncommitted changes
```

## 2. *Créer la branche feature*

Ajouter sur les fichiers **index.html** et **index.js** une première modification quelconque et le commiter avec le message "1".

Ajouter sur le fichier **index.html** et **index.js** une deuxième modification quelconque et le commiter avec le message "2".

Ajouter sur le fichier **index.html** et **index.js** une troisième modification quelconque et le commiter avec le message "3".

## 3. *Retour sur la branche master*

Ajouter sur le fichier **index.html** une première modification quelconque et le commiter avec le message "D".

## 4. *Rebase de feature par rapport à la branche master*

Faire un rebase de la branche **feature** par rapport à la branche **master**.

## 5. *Merge de feature sur la branche master*

Merger la branche **feature** sur la branche **master**.

## VII) TP avec ECommerceSiteGitFlow

### 1. Créer le dossier *Git/tp/ECommerceSiteGitFlow*

Il contiendra notre application de e-commerce. Chaque module contiendra un fichier **index.html** et un fichier **index.js**.

Chaque module de notre site de e-commerce sera développé sous forme de branche.

Nous aurons les modules ci-dessous :

User  
Role  
Catalog  
Basket  
Payment  
Shipping

### 2. Créer les branches pour les différents modules

Créer la branche **develop** et à partir de cette dernière créer une branche pour chaque module c'est-à-dire **feature-1-user**, **feature-1-role**, **feature-1-catalog**, **feature-1-basket**, **feature-1-payment** et **feature-1-shipping**.

### 3. Mise en production : scénario 1

Votre client vous demande une mise en production de l'ensemble des fonctionnalités donc réaliser les opérations nécessaires pour la réussite de cette mise en production.

### 4. Mise en production : scénario 2

Créer un deuxième lot de branches pour chaque module c'est-à-dire **feature-2-user**, **feature-2-role**, **feature-2-catalog**, **feature-2-basket**, **feature-2-payment** et **feature-2-shipping**.

Votre client décide cependant d'amener en production que les branches **feature-2-user**, **feature-2-role**, **feature-2-catalog** et **feature-2-basket**. Il a décidé que les branches **feature-2-payment** et **feature-2-shipping** seront déployés lors d'une autre mise en production. Réaliser les opérations nécessaires pour la réussite de cette mise en production.