TP-Flow

Lien github : https://github.com/AntoninMasson/TP-Flow

Tout d'abord, on crée la branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git checkout -b develop
Switched to a new branch 'develop'
```

Ensuite, on crée la branch feature01 dans la branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git checkout -b feature01 develop
Switched to a new branch 'feature01'
```

On crée notre fichierA que l'on va add puis commit :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git add fichierA

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git commit -m "Ajout fichierA"
[feature01 6c635c8] Ajout fichierA
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierA
```

Le commit sert à sauvegarder des modifications effectuées.

J'ai effectué un premier push pour vérifier que tout fonctionne correctement, le push sert à envoyer nos commit à notre référentiel distant :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git push origin feature01
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 288 bytes | 288.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature01' on GitHub by visiting:
remote:      https://github.com/AntoninMasson/TP-Flow/pull/new/feature01
remote:
To https://github.com/AntoninMasson/TP-Flow.git
 * [new branch]      feature01 -> feature01
```

Une fois que cela est fait, on crée une branch hotfix dans la branch main déjà existante :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git checkout -b hotflx main
Switched to a new branch 'hotflx'
```

Nous allons ensuite créer notre fichierD dans la branch develop, pour cela, on se rend dans la branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (hotflx)
$ git checkout develop
Switched to branch 'develop'
```

Puis on crée le fichierD :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git add fichierD
```

Comme pour chaque ajout de fichier, nous allons commit avec un nom qui correspond :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git commit -m "Creation fichierD"
[develop 6c95f68] Creation fichierD
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierD
```

Désormais, on crée la branch feature02 :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git checkout -b feature02 develop
Switched to a new branch 'feature02'
```

Et on y crée le fichierC :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature02)
$ git add fichierC

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature02)
$ git commit -m "Creation fichierC"
[feature02 fcaff14] Creation fichierC
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierC
```

Dans notre brauch hotfix, on crée le fichierZ :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature02)
$ git checkout hotflx
Switched to branch 'hotflx'

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (hotflx)
$ git add fichierZ

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (hotflx)
$ git commit -m "Creation fichierZ"
[hotflx bc53c14] Creation fichierZ
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierZ
```

Puis on va fusionner hotflx à notre brauch main. Pour ce faire, on se déplace dans la brauch main puis on effectue la commande suivante en précisant la brauch qu'on veut merge :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (hotflx)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git merge hotflx
Updating 38c5c33..bc53c14
Fast-forward
 fichierZ | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierZ
```

Pour chaque fusion vers la brauch main, on tag les commits, on précise la version puis le nom :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git tag -a v1.1 -m "Ajout fichierZ"
```

Puis on commit :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git commit -m "Merge hotfix"
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

Comme la première fois, nous allons vérifier que tout fonctionne correctement et on effectue un push pour vérifier le résultat :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 290 bytes | 290.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AntoninMasson/TP-Flow.git
   38c5c33..bc53c14  main -> main
```

Après avoir modifié notre fichierD dans la branch develop, on le commit instantanément comme pour la création :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git add fichierD

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git commit -m "FichierD modifié"
[develop c5bc56e] FichierD modifi
 1 file changed, 1 insertion(+)
```

Après que le fichierD soit modifié, on merge la branch hotfix vers la branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git merge hotflx
Merge made by the 'ort' strategy.
 fichierZ | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierZ
```

Dans la branch feature01, on va créer le fichierB :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git checkout feature01
Switched to branch 'feature01'

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git add fichierB

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git commit -m "Creation fichierB"
[feature01 7143ca5] Creation fichierB
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierB
```

Ensuite on créé notre dernière branch, la branch release, dans la branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (feature01)
$ git checkout -b release develop
Switched to a new branch 'release'
```

On effectue également notre dernière modification sur le fichierD et on le commit :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (release)
$ git add fichierD

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (release)
$ git commit -m "FichierD encore modifié"
[release bca7e36] FichierD encore modifi
 1 file changed, 2 insertions(+), 1 deletion(-)
```

On crée le fichierE, qui est le dernier, dans notre branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (release)
$ git checkout develop
Switched to branch 'develop'

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git add fichierE

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git commit -m "Creation fichierE"
[develop b4c5cd8] Creation fichierE
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierE
```

On merge la branch feature02 dans un premier temps et dans un second temps la branch feature01 vers notre branch develop :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git merge feature02
Merge made by the 'ort' strategy.
 fichierC | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierC
```

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git merge feature01
Merge made by the 'ort' strategy.
 fichierA | 0
 fichierB | 0
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fichierA
 create mode 100644 fichierB
```

On merge également la branch release :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git merge release
Merge made by the 'ort' strategy.
 fichierD | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)
```

On merge une nouvelle fois la branch release, mais cette fois-ci vers la branche main :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (develop)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git merge release
Updating bc53c14..bca7e36
Fast-forward
 fichierD | 2 ++
 1 file changed, 2 insertions(+)
 create mode 100644 fichierD
```

Comme la première fois, on oublie pas de tag la fusion vers la branch main :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git tag -a v1.2 -m "Ajout fichierD"
```

Pour finir on push toutes les branch une par une :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (10/10), 1.12 KiB | 573.00 KiB/s, done.
Total 10 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/AntoninMasson/TP-Flow.git
   bc53c14..bca7e36  main -> main

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push origin feature01
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 287 bytes | 287.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AntoninMasson/TP-Flow.git
   6c635c8..7143ca5  feature01 -> feature01

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push origin feature02
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 284 bytes | 284.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature02' on GitHub by visiting:
remote:      https://github.com/AntoninMasson/TP-Flow/pull/new/feature02
remote:
To https://github.com/AntoninMasson/TP-Flow.git
 * [new branch]      feature02 -> feature02
```

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push origin release
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'release' on GitHub by visiting:
remote:      https://github.com/AntoninMasson/TP-Flow/pull/new/release
remote:
To https://github.com/AntoninMasson/TP-Flow.git
 * [new branch]      release -> release

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push origin hotflx
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'hotflx' on GitHub by visiting:
remote:      https://github.com/AntoninMasson/TP-Flow/pull/new/hotflx
remote:
To https://github.com/AntoninMasson/TP-Flow.git
 * [new branch]      hotflx -> hotflx

anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push origin develop
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 1022 bytes | 1022.00 KiB/s, done.
Total 8 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), done.
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:      https://github.com/AntoninMasson/TP-Flow/pull/new/develop
remote:
To https://github.com/AntoninMasson/TP-Flow.git
 * [new branch]      develop -> develop
```

On push les tags :

```
anton@PC-ANTONIN MINGW64 ~/Documents/CoursLC/CI-CD/TP-Flow (main)
$ git push origin --tags
Enumerating objects: 2, done.
Counting objects: 100% (2/2), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 311 bytes | 311.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/AntoninMasson/TP-Flow.git
 * [new tag]         v1.1 -> v1.1
 * [new tag]         v1.2 -> v1.2
```