

TD 2d

Un dépôt git pour la recette

Objectif : versionner la recette du TD 2c, en suivre les versions, les différences et les branches, et résoudre un conflit de fusion

`cours2/2d_depot_recette/` $\approx 40'$

Étapes 0 et 1 : le dépôt et le premier commit

	Ce qu'il faut faire	Ce que vous constatez
0	<code>copier</code> <code>recette.md</code> et <code>crepes.jpg</code> du TD 2c dans <code>2d_depot_recette/travail/</code> ; ouvrir <code>travail/</code> dans VS Code	le terminal de VS Code s'ouvre dans <code>travail</code>
1	<code>git init</code> , puis <code>ls -a</code>	
1	<code>git config user.name "Prénom Nom"</code> , puis <code>git config user.email ...</code>	
1	<code>git status</code>	
1	<code>git add recette.md crepes.jpg</code> , <code>git commit -m "Ajoute la recette des crêpes"</code>	

Sans `recette` au TD 2c : `cp ../depart/recette.md ../depart/crepes.jpg . depuis` `travail/`.

Étapes 2 et 3 : modifier, comparer, restaurer

	Ce qu'il faut faire
2	passer les œufs de 4 à 3 ; <code>git status</code> , puis <code>git diff</code>
2	<code>git add recette.md</code> , <code>git commit -m "Réduit les œufs à trois"</code> , <code>git log --oneline</code>
3	abîmer une ligne de la préparation, enregistrer ; <code>git diff</code>
3	<code>git restore recette.md</code> , puis <code>git status</code>

Étape 4 : des fichiers produits par pandoc

	Ce qu'il faut faire
4	<code>pandoc recette.md -o recette.html, puis -o recette.odt ; git status</code>
4	<code>un fichier .gitignore avec deux lignes, *.html et *.odt ; git status</code>
4	<code>git add .gitignore, git commit -m "Ignore les fichiers produits par pandoc"</code>

Étape 5 : une branche, une fusion

	Ce qu'il faut faire
5	<code>git checkout -b sans-gluten ; farine de sarrasin ; commit</code>
5	<code>git checkout master ; ajouter une section « Conseil » ; commit</code>
5	<code>git merge sans-gluten -m "Fusionne la variante sans gluten"</code>

Étape 6 : un conflit, puis le graphe du dépôt

	Ce qu'il faut faire
6	branche <code>pour-18</code> : 18 crêpes ; sur <code>master</code> : 2 heures de repos, sur la même ligne
6	<code>git merge pour-18</code>
6	écrire la bonne ligne, retirer les marqueurs ; <code>git add</code> , <code>git commit</code>
6	dessiner sur papier le graphe du dépôt, puis <code>git log --oneline --graph --all --decorate</code>

Pour aller plus loin : l'étape 7 du guide, une seconde copie du dépôt par `git clone`, puis `git pull`.