

Prénom : \_\_\_\_\_

Nom : \_\_\_\_\_

## Initiation à Git : questionnaire

Entourez la ou *LES* bonnes réponses.

### Question 1

Qu'est-ce qu'un commit ?

- A. Une étape validée du code qui apparaît dans l'historique du dépôt.
- B. L'un des multiples historiques contenus dans un dépôt git.
- C. N'importe quelle modification récente faite sur un des fichiers du dépôt.

### Question 2

Comment connaître l'état courant d'un dépôt ?

- A. `git add`
- B. `git checkout`
- C. `git status`
- D. `git branch`

### Question 3

Qu'est-ce que HEAD ?

- A. Le serveur où on pousse son code.
- B. Un curseur pointant sur un commit qu'on peut déplacer avec `git checkout`.
- C. Une interface pour utiliser git.

### Question 4

Une forge logicielle comme framagit est :

- A. Une plateforme alternative qui remplace git.
- B. Une plateforme pour partager du code en ligne.
- C. Une plateforme de tutoriels pour apprendre la programmation.
- D. Une plateforme utile pour collaborer en entreprise.

### Question 5

Git permet de :

- A. Gérer plusieurs versions du code d'un logiciel/script
- B. Corriger automatiquement du code
- C. Explorer l'historique du code d'un logiciel
- D. Obtenir de l'aide sur la syntaxe en Python

### Question 6

Où sont cachés les versions précédentes des fichiers dans git ?

- A. Dans Gitlens.
- B. Dans un dossier invisible .git pour chaque dépôt.
- C. Dans le dossier /etc.

### Question 7

Pour changer de branche on utilise:

- A. `git reflow <nom de la branche>`
- B. `git checkout <nom de la branche>`
- C. `git clone <nom de la branche>`

### Question 8

Habituellement, quel est le nom de la branche principale d'un dépôt ?

- A. `master`
- B. `feature`
- C. `main`

### Question 9

Une branche est :

- A. Une nouvelle modification ajoutée à un dépôt
- B. Une ligne d'historique du dépôt
- C. Une opération de fusion

### Question 10

Comment savoir quelles modifications ont été apportées lors du dernier commit ?

- A. Utiliser `git diff HEAD HEAD~1`.
- B. Utiliser **gitLens** dans VSCode avec la vue historique.
- C. Utiliser *Thonny* pour debugger le code.

### Question 11

Une merge request sert à :

- A. Créer une discussion avec des collègues sur le code d'une fonction.
- B. Faciliter la vérification collaborative du code.
- C. Corriger automatiquement le code de votre nouvelle fonction.
- D. Tester automatiquement le code ajouté.