Comprehensive Git Interview Questions

Scenario: You want to check which branch you are currently working on.

Answer:

Use:

```
git branch
```

Example:

```
$ git branch
* main
  feature-login
```

Scenario: How do you remove a file from being tracked by Git but keep it in your working directory?

Answer:

Use:

```
git rm --cached <file-name>
```

Example:

```
git rm --cached config.json
```

Scenario: How do you rename a branch?

Answer:

To rename the current branch:

```
git branch -m <new-branch-name>
```

To rename another branch:

```
git branch -m <old-branch-name> <new-branch-name>
```

Scenario: How do you reset a file to its last committed state?

Answer:

Use:

```
git checkout -- <file-name>
```

Example:

```
git checkout -- index.html
```

Scenario: How do you create a patch for a specific commit?

Answer:

Use:

```
git format-patch -1 <commit-hash>
```

Example:

```
git format-patch -1 abcd1234
```

Scenario: You mistakenly committed sensitive information. How do you remove it from all commit history?

Answer:

Use the git filter-repo command:

```
git filter-repo --path <file-name> --invert-paths
```

Example:

```
git filter-repo --path secrets.txt --invert-paths
```

Scenario: How do you handle a detached HEAD state?

Answer:

You can create a new branch from the detached HEAD:

```
git checkout -b <new-branch-name>
```

Example:

```
git checkout abcd1234
```

git checkout -b temp-branch

Scenario: How do you reset a branch to match the remote branch completely?

Answer:

Use:

git fetch origin

git reset --hard origin/<branch-name>

Example:

git reset --hard origin/main

Scenario: You need to rewrite the commit history of a branch. How do you do it?

Answer:

Use interactive rebase:

git rebase -i HEAD~<number-of-commits>

Example:

To rewrite the last 3 commits:

git rebase -i HEAD~3

Scenario: How do you squash multiple commits into one before merging a branch?

Answer:

Use:

git rebase -i HEAD~<number-of-commits>

Example:

To squash the last 3 commits:

git rebase -i HEAD~3

Then edit the commit message as prompted.

Scenario: How do you ensure code quality in a collaborative environment before merging a branch?

Answer:

Set up branch protection rules in your Git hosting platform (e.g., GitHub). Require pull requests and

mandatory code reviews before merging.

Scenario: A teammate accidentally pushed a large file to the repository. How do you remove it to reduce repository size?

Answer:

Use the BFG Repo-Cleaner:

```
bfg --delete-files <file-pattern> <repo-path>
```

Example:

To remove all `.zip` files:

```
bfg --delete-files "*.zip" my-repo.git
```

Scenario: Your build server requires a specific tag to deploy. How do you create and push a tag?

Answer:

Use:

```
git tag <tag-name>
git push origin <tag-name>
```

Example:

```
git tag v1.0.0
```

git push origin v1.0.0

Scenario: You want to make an experimental change without affecting the main branch. How do you handle it?

Answer:

Create a new branch for experimentation:

```
git checkout -b experiment-branch
```

Work on the branch and merge it if successful.