1.

What is the difference between comparing two branches with git diff or git difftool?

* Incorrect -

git diff is read-only, while git difftool opens the diff in interactive mode.

* Your choice: correct -

git diff uses the command-line diff command, while git difftool uses the external diff tool that you configured.

* Incorrect -

git diff uses vim style code highlighting, while git difftool does not provide code highlighting at all.

* Incorrect -

git diff uses the system's default diff program, while git difftool allows you to specify the name of a diff program.

* Incorrect -

I don't know yet.

2.

Which command creates an annotated tag named mytag on the current commit?

* Incorrect -

git annotate mytag

* Your choice: correct -

git tag -a mytag

* Incorrect -

git tag mytag

* Incorrect -

git tag mytag "my comment"

* Incorrect -

I don't know yet.

3.

How do you rebase the current branch (master) over branch1?

* Incorrect -

git rebase master -b branch1

* Incorrect -

git rebase master

* Incorrect -

git rebase master branch1

* Your choice: correct -

git rebase branch1

* Incorrect -

I don't know yet.

4.

Git stops in the middle of a merge because a file was modified in both branches. How would you determine which file caused the conflict?

* Your choice: correct -

The git status command lists the conflicting file as "Both modified."

* Incorrect -

The git log command lists the conflicting file as "Conflicting."

* Incorrect -

The git status command lists the conflicting file as "Conflicting."

* Incorrect -

You must manually analyze the files in both branches to locate the conflict.

* Incorrect -

I don't know yet.

5.

What is the difference between the bare flag and mirror flag when cloning a repository?

* Incorrect -

They will both clone the repository without a working directory. Mirror will automatically update the local repository so it reflects the remote repository.

* Your choice: incorrect -

They will both clone the repository with a working directory. Mirror will also set up tracking for all of the remote references.

* Correct -

They will both clone the repository without a working directory. Mirror will also set up tracking for all of the remote refs.

* Incorrect -

They will both clone the repository with a working directory. Bare will also set up tracking for all of the remote references.

* Incorrect -

I don't know yet.

6.

Which statement best explains the purpose of git show?

* Incorrect -

It shows the commit history for the current active branch.

* Your choice: correct -

It shows any Git object in a human-readable format.

* Incorrect -

It shows the modified files in the working directory.

* Incorrect -

It shows a list of stashed file changes.

* Incorrect -

I don't know yet.

7.

You need to undo a merge commit. What information will you need to locate and include in the command?

* Incorrect -

The SHA-1 for each parent branch

* Incorrect -

A reference to the git resolution used to resolve conflicts in the merge

* Your choice: correct -

The parent number associated with the desired mainline branch

* Incorrect -

The name of a new branch where the reverted commit will be applied

* Incorrect -

I don't know yet.

8.

You want to remove file2.txt from the index. Which sequence of commands is git rm a shorthand for in this situation?

* Incorrect -

First rm file2.txt, then git commit -m "Removing file2.txt"

* Your choice: correct -

First rm file2.txt, then git add file2.txt

* Incorrect -

First rm file2.txt, then git add file2.txt, then git commit -m "Removing file2.txt"

* Incorrect -

First git add file2.txt, then rm file2.txt

* Incorrect -

I don't know yet.

9.

What does rerere mean?

* Your choice: correct -

Reuse Recorded Resolution

* Incorrect -

Replace Regex Recursively

* Incorrect -

Restore Remote Relay

* Incorrect -

Replay Responsive Rebase

* Incorrect -

I don't know yet.

10.

What does the following command accomplish?

`git filter-branch --env-filter '

if test "$GIT\_AUTHOR\_EMAIL" = "admin@localhost"

then

GIT\_AUTHOR\_EMAIL=john@pluralsight.com

fi

' -- --all`

* Incorrect -

It creates a new branch that uses John's email address as commit author instead of the Admin email address.

* Incorrect -

It filters all commits within the repository where Admin is set as the Git author and outputs them to a log file for further inspection.

* Your choice: incorrect -

It updates the metadata of a commit within a branch to use John's email address as commit author instead of the Admin email address.

* Correct -

It updates the metadata of each commit in the repository to use John's email address as commit author instead of the Admin email address.

* Incorrect -

I don't know yet.

11.

Which of the following can result in the creation of more than one commit?

* Incorrect -

Neither rebase nor merge

* Your choice: correct -

rebase

* Incorrect -

Both rebase and merge

* Incorrect -

merge

* Incorrect -

I don't know yet.

12.

What will the following Git command output?

git log -20 --after="2030-4-1" --before="2030-4-30" --oneline --author="Dev1"

* Your choice: correct -

This will list at most 20 commits by any developer named Dev1 in descending order from April 1, 2030 to April 30, 2030.

* Incorrect -

This will list at least 20 commits by any developer named Dev1 starting from April 30, 2030 to April 1, 2030.

* Incorrect -

This will list at most 20 commits by any developer named Dev1 in ascending order from April 1, 2030 to April 30, 2030.

* Incorrect -

This will list at least 20 commits by any developer named Dev1 starting from April 1, 2030 to April 30, 2030.

* Incorrect -

I don't know yet.

13.

You have four branches, each of which has its own commits that are different from the other branches. You then perform a single git merge instruction to merge all four branches together. How many commits are created?

* Incorrect -

4

* Incorrect -

3

* Correct -

1

* Your choice: incorrect -

0 or 1

* Incorrect -

I don't know yet.

14.

What is the significance of --- versus +++ on file1.txt below?

git diff a/file1.txt b/file1.txt

index 94ec7fe..81cc76f 100644

--- a/file1.txt

+++ b/file1.txt

* Incorrect -

The +++ indicates the prior version and --- indicates the version with the latest changes.

* Incorrect -

The +++ shows whether the file was moved and --- shows the prior location.

* Incorrect -

The +++ indicates a new file that was added and --- indicates a recent file that was deleted.

* Your choice: correct -

The +++ indicates the version with the latest changes and --- indicates the prior version.

* Incorrect -

I don't know yet.

15.

Can you have a commit that does not point to any tree?

* Incorrect -

Yes, if it is the first commit in the project.

* Your choice: correct -

No, a commit always points to a tree, which represents the root directory of the project.

* Incorrect -

No, a commit must point to the same tree as its parent commit.

* Incorrect -

Yes, if the commit is pointing to another commit.

* Incorrect -

I don't know yet.

16.

During a rebase, which command would you use to mark a conflict as resolved?

* Incorrect -

git resolve

* Incorrect -

git rebase --proceed

* Your choice: correct -

git add

* Incorrect -

git commit

* Incorrect -

I don't know yet.

17.

A submodule contains changes, including local changes and upstream changes from various contributors. You also have changes ready for commit in the main project. What is the best solution to ensure both main project and submodule changes are available for all contributors to access?

* Incorrect -

git push --recurse-submodules=check

* Incorrect -

git push --recurse-submodules=safe

* Correct -

git push --recurse-submodules=on-demand

* Your choice: incorrect -

git push --recurse-submodules=serial

* Incorrect -

I don't know yet.

18.

You accidentally reset a developer's current working directory and use git reset --hard HEAD~1, which wipes out the developer's last commit and working directory changes. How could you undo the reset and revert the situation?

* Incorrect -

Use git reflog -i 0 to get the hash of the latest changed commit and restore it.

* Your choice: correct -

Use git reflog to get the hash of the commit you want to restore, and then run git merge <hash>. There is a time limit of 90 days, however.

* Incorrect -

Use the git reset --restore <hash of your commit> to undo the reset and revert the situation .

* Incorrect -

Use git reflog to get the location of the commit you want to restore.

* Incorrect -

I don't know yet.

1.

You observe that the team primarily resolves merge conflicts by navigating the standard conflict markers, such as, <<<<<<<, =======, and >>>>>>>, but uses graphical user interface (GUI) applications for all other tasks. What is the most likely explanation for this discrepancy?

* Incorrect -

The team performed a rebase on a project, which only allows merges.

* Your choice: correct -

The team is not aware of merge utilities, which are available with git mergetool.

* Incorrect -

The team has not configured its merge conflict settings with git rerere.

* Incorrect -

The team has a pre-commit hook configured to add the markers into the files for quality control.

* Incorrect -

I don't know yet.

2.

What does the ^ in git show HEAD^ denote?

* Your choice: correct -

The parent of

* Incorrect -

The index of

* Incorrect -

The child of

* Incorrect -

The remote copy of

* Incorrect -

I don't know yet.

3.

What purpose does the cached flag provide for git diff?

* Incorrect -

It is the same functionality as git diff.

* Your choice: incorrect -

It compares changes in the latest stash with the current uncommitted changes.

* Incorrect -

It compares changes that are not staged yet with the local repository.

* Correct -

It compares the staged changes with the local repository version.

* Incorrect -

I don't know yet.

4.

Which command creates an annotated tag named mytag on the current commit?

* Incorrect -

git annotate mytag

* Incorrect -

git tag mytag

* Incorrect -

git tag mytag "my comment"

* Your choice: correct -

git tag -a mytag

* Incorrect -

I don't know yet.

5.

What does git grep accomplish?

* Incorrect -

It lists the commits that contain certain patterns.

* Incorrect -

It lists the authors that wrote specified patterns.

* Your choice: correct -

It searches for specific patterns in your repository.

* Incorrect -

There is a grep command, but no git grep command.

* Incorrect -

I don't know yet.

6.

When is it necessary to use Git's low-level (or "plumbing") commands?

* Incorrect -

When you are debugging missing changes from a merge conflict

* Incorrect -

When you are instrumenting organization-wide configurations

* Incorrect -

When you are optimizing for performance across multiple repositories

* Your choice: correct -

When you are building new Git tools or creating intricate scripts

* Incorrect -

I don't know yet.

7.

Which types of objects can be part of the Git object database?

* Incorrect -

Commits, Blobs, and Trees

* Incorrect -

Commits, Blobs, Trees, and Non-annotated Tags

* Incorrect -

Commits, Blobs, Trees, and Branches

* Your choice: correct -

Commits, Blobs, Trees, and Annotated Tags

* Incorrect -

I don't know yet.

8.

You are setting up a server and would like to add your repository for running continuous integration pipelines. This clone must map all references to their target branches whether they're local or remote. In addition, the local changes must be overwritten by a call to git remote update. Which command will set up this repository correctly?

* Incorrect -

git clone --all-branches

* Incorrect -

git clone

* Incorrect -

git clone --bare

* Your choice: correct -

git clone --mirror

* Incorrect -

I don't know yet.

9.

When performing a revert on a single commit, you encounter the following error:

Error: your local changes would be overwritten by revert.

Which method of resolution would not permit you to perform the revert operation?

* Your choice: correct -

Continue Revert

* Incorrect -

Commit Changes

* Incorrect -

Discard Changes

* Incorrect -

Stash Changes

* Incorrect -

I don't know yet.

10.

Using the following stash list that has a last in, first out (LIFO) stashing data structure, how do you pull in the oldest stash while ensuring that you maintain it in the stash?

stash@{0}: WIP on feature/feature1: 850f537 New text

stash@{1}: WIP on feature/feature1: 850f537 New text

* Incorrect -

git stash pop stash@{1}

* Your choice: correct -

git stash apply stash@{1}

* Incorrect -

git stash apply stash@{0}

* Incorrect -

git stash pop stash@{0}

* Incorrect -

I don't know yet.

11.

What command allows you to delete the remote branch feature/test?

* Incorrect -

git branch -d --remote feature/test

* Correct -

git push origin :feature/test

* Incorrect -

git branch :feature/test

* Your choice: incorrect -

git branch -d origin feature/test

* Incorrect -

I don't know yet.

12.

During a rebase, Git executes a fast-forward. What would have occurred if you chose merge instead, assuming no force option was set?

* Incorrect -

That scenario is unlikely because due to Git's inability to fast-forward a rebase.

* Incorrect -

It depends on the specific scenario because a merge may or may not have resulted in a fast-forward.

* Correct -

Git would execute a fast-forward as there is no difference between merge and rebase in that regard.

* Your choice: incorrect -

The merge would create a new commit, which makes the project's history harder to read.

* Incorrect -

I don't know yet.

13.

Which two commands, from the options below, would you use together to output all commits from the last two weeks?

* Incorrect -

git log --before="14 days ago" and git log --before="now"

* Incorrect -

git log --after="2" --unit="weeks" and git log --after="2" --unit="weeks"

* Your choice: correct -

git log --after="2 weeks ago" and git log --after="14 days ago"

* Incorrect -

git log --before="2 weeks ago" and git log --after="14 days ago"

* Incorrect -

I don't know yet.

14.

At the command git log master, Git displays an error. Which scenario caused the error?

* Incorrect -

You are not on the master branch.

* Incorrect -

You are in a detached HEAD situation.

* Incorrect -

The command will not work as you cannot pass a branch name to git log.

* Your choice: correct -

The master branch does not exist, or it does not have a commit.

* Incorrect -

I don't know yet.

15.

What will the following Git command output?

git log -20 --after="2030-4-1" --before="2030-4-30" --oneline --author="Dev1"

* Incorrect -

This will list at most 20 commits by any developer named Dev1 in ascending order from April 1, 2030 to April 30, 2030.

* Your choice: correct -

This will list at most 20 commits by any developer named Dev1 in descending order from April 1, 2030 to April 30, 2030.

* Incorrect -

This will list at least 20 commits by any developer named Dev1 starting from April 30, 2030 to April 1, 2030.

* Incorrect -

This will list at least 20 commits by any developer named Dev1 starting from April 1, 2030 to April 30, 2030.

* Incorrect -

I don't know yet.

16.

Which commits are removed when utilizing the command git reset --hard HEAD~3?

* Incorrect -

HEAD~3

* Incorrect -

HEAD~1, HEAD~2

* Incorrect -

HEAD~1, HEAD~2, and HEAD~3

* Your choice: correct -

HEAD, HEAD^, and HEAD~2

* Incorrect -

I don't know yet.

17.

You stash your work, but you continue making changes to the files over several commits. What is the significance of the following:

git stash branch add-feature stash@{2}

* Incorrect -

An interactive rebase will start so you can safely modify the commit stack between the current branch and the stash that you are applying.

* Incorrect -

This will throw an error. Stash is a simple tool and cannot work with branches.

* Incorrect -

Stash will create a new branch and attempt to merge the latest changes with the stashed changes.

* Correct -

Stash can create a new branch at the commit the stash was created in order to cleanly apply the stash with no conflicts.

* Incorrect -

I don't know yet.

18.

How many commits are created if you merge two branches?

* Your choice: correct -

0 or 1

* Incorrect -

0

* Incorrect -

1

* Incorrect -

2

* Incorrect -

I don't know yet.

1. You tag the current branch and then rename it. Does that result in an error?

Incorrect -Yes, you would break the reference in the tag by renaming the commit.

//Correct -No, the tag points to the commit, not the branch.

Incorrect -No, Git will update the tag to point to the same branch even if it is renamed.

Incorrect -No, the tag is garbage-collected.

Incorrect -I don't know yet.

2. You are working on an open source project that is growing in size, complexity, and the number of maintainers. You must improve the current workflow, which is standard GitFlow. How could you make improvements?

Incorrect -Restrict who can push to the stable branches to only the original author of the project.

Incorrect -Innovate unique branch naming conventions that are less ambiguous.

//Correct -Establish more frequent integration between the feature and integration branches.

Incorrect -Set a biweekly deadline for merges into master or other integration branches.

Incorrect -I don't know yet.

3. What are some disadvantages to using Git with the SSH transport protocol?

Incorrect -All information is insecurely transmitted in plain text and it requires recurring payments for usage fees.

//Correct -It does not allow anonymous access and involves a more complicated set up process for contributors.

Incorrect -It is difficult to set up due to the utilization of unstandard ports, which cannot provide authentication.

Incorrect -There is limited location access and it is not the fastest option.

Incorrect -I don't know yet.

4. Which command provides you with a console to quickly diff, patch, revert, and update files?

//Correct -git add --interactive

Incorrect -git add --patch

Incorrect -git add

Incorrect -git add --update

Incorrect -I don't know yet.

5. While on branch master, you give the command git merge branch1. Which branch or branches change?

Incorrect -branch1

Incorrect -Neither

//Correct -master

Incorrect -Both

Incorrect -I don't know yet.

6. You want to create a new branch called mybranch, but you already have an old branch with that name. You want to safely delete the old branch. Which of these operations would solve the problem with the least effort?

Incorrect -Checkout the existing branch to the current commit

//Correct -Delete the old branch, then create the new branch

Incorrect -Create the new branch and Git will automatically replace the old branch with the same name

Incorrect -None of these operations will solve the problem, and you cannot create the new branch unless you select a different name

Incorrect -I don't know yet.

7. What does a newly created branch point to?

Incorrect -The current branch

Incorrect -The last change staged

//Correct -The current HEAD

Incorrect -A new commit

Incorrect -I don't know yet.

8. Which command cannot result in a conflict?

Incorrect -git cherry-pick

//Correct -git fetch

Incorrect -git rebase

Incorrect -git merge

Incorrect -I don't know yet.

9. You made a typo in your most recent commit message and have not yet pushed to remote. How would you correct it?

Incorrect -git amend {SHA-1}

Incorrect -git reset {SHA-1}

//Correct -git commit --amend

Incorrect -git reset --soft HEAD

Incorrect -I don't know yet.

10. What kind of repository is created with the git init --bare command?

//Correct -A bare repository that lacks a working directory

Incorrect -A centralized repository, with multiple working directories

Incorrect -An installed repository and the files created during the install process will need to be pruned from the working directory

Incorrect -A replay repository that will accept any changes pushed to it in its working directory.

Incorrect -I don't know yet.

11. What is the repository in a version control system?

//Correct -A data structure that stores references to a project's files and folders, including their history

Incorrect -A data structure that contains the configuration of a version control system, including the names of files and folders

Incorrect -A data structure that represents the only authoritative copy of a distributed project's history

Incorrect -A data structure that contains the only shared copy of the current files and folders in a project

Incorrect -I don't know yet.

12. What is the benefit of using the graph flag in the command git log --oneline --graph --decorate --all?

Incorrect -It simplifies the output by using an ASCII graph to show only the flow of the local branches.

Incorrect -It does nothing since it is not a flag for git log; it is a flag for git blame.

//Correct -It simplifies the output by using an ASCII graph to show the flow of the different branches in the repository.

Incorrect -It removes the commit IDs in the log, which produces ease when viewing commits.

Incorrect -I don't know yet.

13. Which URL protocols would you use when fetching a git repository?

Incorrect -SSH, Git, and UDP

//Correct -SSH, Git, and HTTP

Incorrect -SSH, TCP, and UDP

Incorrect -SSH, Git, and TCP

Incorrect -I don't know yet.

14. Which command both creates a new commit and undoes a previous commit?

Correct -git revert

Incorrect -git merge

Incorrect -git append -d

Incorrect -git mv

Incorrect -I don't know yet.

15. If a hotfix is made on a release (stable) branch, what actions are necessary for other branches, i.e., integration, feature, etc.?

//Correct -The changes must be replicated by rebasing, cherry-picking, or merging.

Incorrect -The branch must be deleted and remade utilizing the changes from the stable branch.

Incorrect -All branches are based on the stable branch, so changes made there will be present in other branches.

Incorrect -It is not necessary to bring hotfixes into feature branches because the changes made are irrelevant.

Incorrect -I don't know yet.

16. Why is amending commits after pushing to remote discouraged?

Incorrect -Force pushing an amended version of a commit causes a new branch to be created upstream, which duplicates work

Incorrect -Pushing an amended commit ignores conflicts resolved previously

Incorrect -Pushing a simple change such as an amended message will not cause problems in Git

//Correct -Force pushing an amended commit to a remote will rewrite history, forcing others to manually update their local repository

Incorrect -I don't know yet.

17. Which command recovers commits that are not referenced by a branch or tag?

Incorrect -git log --graph

Correct -git reflog

Incorrect -git describe

Incorrect -git show-branch

Incorrect -I don't know yet.

18. At your office, it is standard practice to delete branches, both local and remote, after merging code from a pull request. You discuss the features built on a particular branch with a colleague, but you cannot look it up in the log as the branches were deleted. Is it possible to find the point in time in which the branch existed, and why?

Incorrect -No, once a branch is deleted after a merge, there is no way to locate where that branch may have existed.

Your choice: incorrect -Yes, by using git branch --deleted, you can pull up all deleted branches and restore them back into Git.

//Correct -Yes, by using git log --merges, you can find only the commits that were merged and look for the one with the branch name in the commit message.

Incorrect -I don't know yet.

1. A rebase interrupts because of a conflict in one of the rebased commits and you skip that specific commit to continue the rebase. How would you accomplish this?

git rebase --skip

2. What happens if you give the command git add .?

You stage the changes in the current directory and, recursively, all the changed files and directories it contains except for empty directories.

3. When is it necessary to use Git's low-level (or "plumbing") commands?

When you are building new Git tools or creating intricate scripts

4. Git implements branches as a running series of commits in a history tree. Which definition would more accurately describe a branch?

A pointer to a commit

5. What does git grep accomplish?

It searches for specific patterns in your repository.

6. How do git rm and rm differ from each other?

rm removes a file, but will not stage the change for the Git repository, whereas git rm can remove the file from either the index or the filesystem, and then stages the change.

7. At the command git log master, Git displays an error. Which scenario caused the error?

The master branch does not exist, or it does not have a commit.

8. You are setting up a server and would like to add your repository for running continuous integration pipelines. This clone must map all references to their target branches whether they're local or remote. In addition, the local changes must be overwritten by a call to git remote update. Which command will set up this repository correctly?

git clone --mirror

9. What does HEAD point to when it is detached?

A commit

10. Git stops in the middle of a merge because a file was modified in both branches. How would you determine which file caused the conflict?

The git status command lists the conflicting file as "Both modified."

11. How do you rebase the current branch (master) over branch1?

git rebase branch1

12. What command can move your working directory to an earlier commit?

git checkout

13. What does the ^ in git show HEAD^ denote?

The parent of

14. Which git log command will produce the following output?

git log --oneline

15. Why is the reflog considered a safety net?

It maintains references to the old state of objects even if they are not connected to a branch.

16. Which command creates an annotated tag named mytag on the current commit?

git tag -a mytag

17. How would you force Git to trigger garbage collection?

With git gc

18. How would you display commit a7855c91ed8c58 (tag: v1.3.1)?

git show a7855 or git show v1.3.1